Learning-Based Calibration, Estimation, and Stage Control

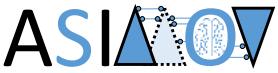
for

Electron Microscopy

JILLES VAN HULST

Learning-Based Calibration, Estimation and Stage Control for Electron Microscopy

Jilles Stefan van Hulst

This research is partly funded by the ITEA4 20216 ASIMOV project. The ASIMOV activities are supported by the Netherlands Organisation for Applied Scientific Research TNO and the Dutch Ministry of Economic Affairs and Climate (project number: AI211006). The research leading to these results is partially funded by the German Federal Ministry of Education and Research (BMBF) within the project ASIMOV-D under grant agreement No. 01IS21022G [DLR], based on a decision of the German Bundestag.

This research is also funded by the research project entitled *Learning in Motion*, a collaboration between the Eindhoven University of Technology and several industry partners. This project is co-financed by Holland High Tech, top sector High-Tech Systems and Materials, with a PPP innovation subsidy for public-private partnerships for research and development.

The work described in this thesis was carried out at the Eindhoven University of Technology.

A catalogue record is available from the Eindhoven University of Technology Library.
ISBN: 978-90-386-6760-7

Typeset by the author using the pdf LaTeX documentation system.
Cover design: Salomé Busurashvili
Reproduction: Ipskamp Printing, Enschede, the Netherlands

Learning-Based Calibration, Estimation and Stage Control for Electron Microscopy

Proefschrift

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus prof. dr. S.K. Lenaerts, voor een commissie aangewezen door het College voor Promoties, in het openbaar te verdedigen op 03-09-2026 om 11:00 uur

door

Jilles Stefan van Hulst

geboren te Veldhoven

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter:	prof. dr. ir. D.M.J. Smeulders	
1e promotor:	dr. D.J. Guerreiro Tomé Antunes	
2e promotor:	prof. dr. ir. W.P.M.H. Heemels	
co-promotor:	dr. ir. E.M. Franken	(Thermo Fisher Scientific)
leden:	prof. dr. K.J. Batenburg	(Leiden Universiteit)
	dr. L. Özkan	
	prof. S. Särkkä	(Aalto University)
	Prof. Dr. S. Trimpe	(RWTH Aachen University)

Het onderzoek dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Summary

Learning-Based Calibration, Estimation and Stage Control for Electron Microscopy

Given noisy observations, what can we learn about the system that produced them? The answer depends not only on the data but also on prior knowledge: a physical model, a symmetry in the data, or a structural constraint on the solution. This interplay between structure and data is the central theme of this thesis, which specifically addresses two persistent challenges. The first is data scarcity: in electron microscopy, every measurement carries a physical cost in dose, time, or operator effort, and thus large labeled datasets are simply not available. The second is the simulation-to-reality gap: while physics-based simulators can provide abundant synthetic training data, simulated and real measurements are generated by different processes, and this discrepancy limits what can be learned from simulation alone. Across eight chapters, we develop machine learning and control methods that exploit known structure to extract more information from less data. The primary application is electron microscopy, one of the most powerful and demanding measurement platforms in modern science, although many of the presented methods apply to various application domains.

Electron microscopes use electrons instead of photons to image matter. Because the wavelength of electrons is far smaller than that of visible light, they can resolve individual atoms, enabling breakthroughs in materials science, structural biology, and semiconductor manufacturing. Achieving such resolution requires these instruments to be precisely tuned and calibrated. Lens imperfections such as aberrations degrade resolution and must be measured and corrected before high-resolution imaging can begin. Furthermore, the specimen must be positioned with nanometer accuracy over millimeter-scale ranges. Existing automation for these tasks relies on simplified physical models or hand-crafted heuristics rather than

general data-driven methods. As instruments grow more complex and throughput demands increase, the limitations of hand-crafted procedures become increasingly apparent, and the field is moving toward data-driven approaches. This thesis contributes to that shift. It develops learning-based methods for the optical calibration and precision positioning of electron microscopes, and it provides general methods for data-efficient machine learning that extend beyond the microscopy application.

The work is organized in three parts. Part I focuses on the calibration of scanning transmission electron microscopes (STEM). The first chapter introduces a data-driven pipeline that combines a convolutional neural network with Bayesian optimization to estimate and correct low-order aberrations from simulated training data. A subsequent chapter advances this pipeline into a principled framework based on variational autoencoders (VAEs). The VAE extracts compact features from simulated microscope images, which are then used to estimate the aberrations. Crucially, this approach succeeds despite the simulation-to-reality gap. By exploiting the symmetry of electron-optical aberrations, an expectation-maximization algorithm can bridge this gap by jointly estimating the aberration state and learning the domain-transfer model.

Part II pertains to the transmission electron microscope (TEM). The first chapter develops a method for estimating TEM aberrations based on images acquired at different beam tilt angles, together with a principled approach to selecting the most informative tilt sequence. The second chapter addresses precision positioning of the specimen stage, which is driven by multiple piezoelectric actuators. These actuators suffer from hysteresis, mechanical misalignments due to manufacturing tolerances, and non-collocated sensing. The proposed method combines data-driven feedforward control with image-based feedback to achieve nanometer-scale positioning accuracy.

Part III broadens its scope beyond the electron microscopy application and contributes to the wider field of data-efficient machine learning. Four chapters address different aspects of this theme: constructing VAE latent spaces with prescribed topological structure, extending Gaussian process regression to time-varying functions, learning optimal control policies from few data by exploiting structural assumptions and linear matrix inequalities, and designing exploration strategies for reinforcement learning based on limited model knowledge.

In every chapter, the key to data efficiency is structure, which constrains the set of admissible solutions. It may take the form of a physical law, a symmetry, a bound on the parameter space, or a dynamical model. This determines which parameters are identifiable from data at all, and it determines how much information each observation carries about them. Where simulation is used, structure constrains the comparison between simulated and real measurements, making the gap bridgeable from limited real data. Rather than relying on large datasets, each method in this

thesis is designed to extract as much information as possible from limited data by formally integrating what is already known. This structured approach to learning provides a robust path forward for complex scientific instruments, particularly when real data is scarce and simulation-driven approaches face the realities of the physical world.

Keywords: Electron microscopy, aberration correction, machine learning, Bayesian estimation, data efficiency, simulation-to-reality gap, variational autoencoders, Gaussian processes, reinforcement learning, iterative learning control, inverse problems, specimen positioning

Contents

1 Introduction

1.1 Electron Microscopy

Electron microscopes use electrons instead of photons to image matter. Because the wavelength of electrons is much smaller than that of visible light, electron microscopes break the diffraction limit of optical microscopes by several orders of magnitude. This makes them capable of resolving individual atoms. Since the first electron microscope was built in 1931, this capability has enabled fundamental discoveries across physics, chemistry, biology, and engineering. Figure 1.1 shows a modern electron microscope. To appreciate where the field stands today, and what challenges remain, it helps to briefly trace how electron microscopy developed. Figure 1.2 summarizes the key milestones.

Ernst Ruska and Max Knoll built the first electron microscope, a transmission electron microscope (TEM), in Berlin in 1931 [2], a breakthrough for which Ruska received the Nobel Prize in Physics in 1986. In the decades that followed, the TEM matured into a versatile tool for materials science and biology. The basic physics of electron optics was understood early on, but achievable resolution was limited by so-called aberrations in the electromagnetic lenses. Aberrations are imperfections in how the lenses focus the electron beam. In 1936, Scherzer proved that any rotationally symmetric electron lens has positive spherical aberration [3], a result known as Scherzer's theorem. Since at the time, all electron lenses were rotationally symmetric, this set a fundamental limit on resolution, and for the next sixty years aberrations remained the primary barrier to atomic-resolution imaging.

A first turning point came in 1970, when Crewe demonstrated single-atom

Figure 1.1. A Thermo Fisher Scientific Spectra scanning/transmission electron microscope, taken from Davies [1]. The column contains the electron gun, multiple electromagnetic lens systems, a phase plate, and a direct electron detector in a housing that exceeds two meters in height. Specimen loading, stage control, and image acquisition are automated, but alignment and calibration of the optical column still require expert intervention before each session.

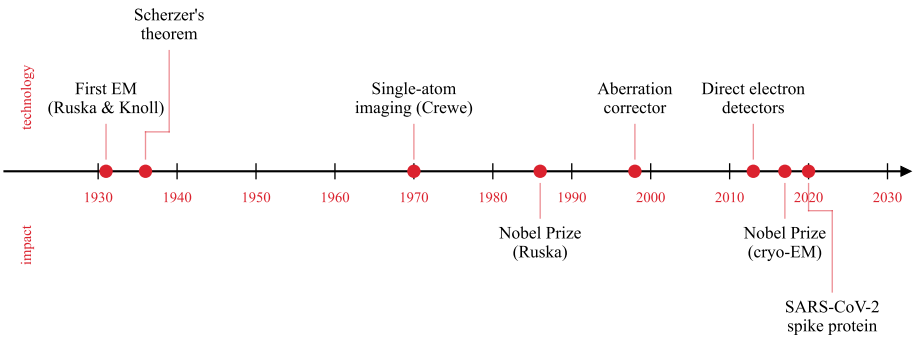
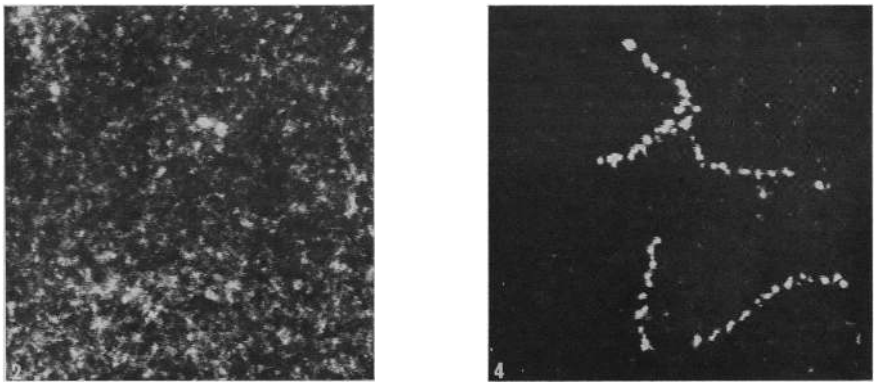


Figure 1.2. Key milestones in electron microscopy. Technology developments are placed above the timeline, and scientific impact and recognition below it.

imaging using a scanning transmission electron microscope (STEM) [4] (Figure 1.3). Rather than improving the lenses, STEM uses a different imaging mode: it rasters



(a) Uranium, $Z = 92$ **(b)** Thorium, $Z = 90$

Figure 1.3. Zoomed dark-field STEM micrographs of single heavy atoms on an amorphous carbon support, reprinted from Crewe et al. [4]. Each bright spot is one atom. The contrast is Z-contrast: both elements scatter far more strongly than the carbon background, making individual atoms visible as discrete spots. The images were obtained without aberration correction, showing that Z-contrast STEM achieved single-atom sensitivity three decades before practical aberration correctors became available.

a focused electron probe across the specimen and collects scattered electrons at each position. This enables Z-contrast imaging, where the image intensity scales with atomic number, and it allows electron energy loss spectroscopy (EELS, where energy lost by inelastic scattering reveals the elemental composition and bonding of the specimen) to be recorded at every probe position. A second and arguably more important turning point came in 1998, when Haider, Rose, and Urban demonstrated the first practical aberration corrector for TEM [5]. By adding multipole lenses, which are not rotationally symmetric and therefore not subject to Scherzer's theorem, the corrector can produce aberrations that cancel those of the objective lens, which substantially improved the point resolution of the TEM. Correctors of a different multipole design were developed for STEM in parallel [6], where they made sub-ångström probes possible [7, 8].

More recently, the 2017 Nobel Prize in Chemistry was awarded to Dubochet, Frank, and Henderson for cryo-electron microscopy (cryo-EM) [9–11], a TEM-based technique in which specimens are preserved by rapid freezing. Imaging at cryogenic temperatures slows radiation damage, enabling the determination of three-dimensional structures of biological molecules at near-atomic resolution. Cryo-EM has since become a standard tool in structural biology and drug discovery. In 2020, it was used to determine the structure of the SARS-CoV-2 spike protein

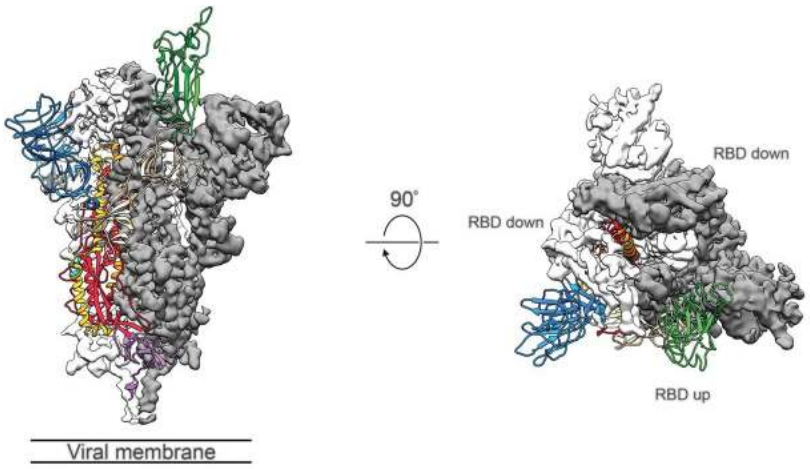


Figure 1.4. Cryo-EM density map of the SARS-CoV-2 spike protein at 3.5 Å resolution, reprinted from Wrapp et al. [12]. The prefusion trimer is shown as a gray surface fitted with an atomic model, in two views (90° apart). The receptor-binding domain (RBD) is visible in both the "up" and "down" conformations.

within weeks of the pandemic's onset, directly contributing to the rapid development of mRNA vaccines [12] (Figure 1.4).

The impact of electron microscopy extends well beyond the instrument itself. Throughout its history, advances in EM hardware have directly enabled scientific results that would otherwise have been out of reach. The introduction of direct electron detectors in the early 2010s triggered what is now called the "resolution revolution" in cryo-EM [13], producing an explosion of protein structure determinations. Similar patterns can be seen in materials science and semiconductor manufacturing, where each new generation of microscopes opens up previously inaccessible research. Today, electron microscopes are indispensable across materials science, semiconductor manufacturing, structural biology, catalysis, and battery research.

As electron microscopes have become more powerful, they have also become more complex. A modern aberration-corrected microscope has dozens of electromagnetic elements that must be tuned simultaneously. The number of parameters, the precision required, and the interactions between components have grown to the point where operating these instruments at their full potential is a significant challenge in itself, which we discuss in more detail in the next section.

1.2 Challenges in Modern Electron Microscopy

The capabilities of EM described above come with open problems that are far from solved. Some are rooted in physics, others in engineering, and many have persisted for decades despite sustained effort. This section surveys the main challenges faced by the field today, focusing on high-resolution TEM and STEM, where atomic-scale information is the goal[1]. Two related areas fall outside the scope of this thesis. The first is scanning electron microscopy (SEM), in which a focused electron beam is scanned across the surface of a specimen and the resulting secondary or backscattered electrons form an image of that surface. SEM has a different challenge profile dominated by surface charging on poorly conducting samples, and the methods in this thesis do not transfer to it directly. The second is specimen preparation, often the rate-limiting step in TEM workflows, which is a separate problem from instrument operation and is likewise not addressed here.

C1: Calibration and Alignment of the Optical Column

The optical column of an electron microscope contains many types of imperfections such as aberrations and distortions that degrade image quality. Fortunately, these imperfections can often be corrected by adjusting the electromagnetic field in precise ways. Aberration correctors have enabled these adjustments and are one of the most transformative hardware advances in the history of electron microscopy, but they come at a cost: complexity. A modern corrector contains dozens of multipole elements that must be tuned to cancel the aberrations of the objective lens. The tuning is not a one-time procedure. Aberrations drift due to thermal effects, specimen charging, and mechanical instabilities, and the corrector must be recalibrated periodically during operation [14].

Calibration is currently often performed or monitored by expert operators who interpret diagnostic diffraction patterns and adjust corrector settings iteratively. In STEM, the most common diagnostic is the Ronchigram. As the converging electron probe passes through the specimen, it forms a diffraction pattern whose spatial structure encodes the current aberration state. The shape and orientation of the diffraction pattern indicate which aberrations are present. On an amorphous specimen, these features are directly readable by an experienced operator. On a crystalline specimen, the same geometry instead produces a convergent-beam electron diffraction (CBED) pattern, in which the standard operator-based analysis breaks down.

In TEM, the standard procedure is the Zemlin tableau [15]. A series of images

[1] On many modern platforms, TEM and STEM are operational modes of a single instrument rather than separate machines. Throughout this thesis, the distinction refers to the imaging mode rather than the hardware platform.

is acquired at different beam tilt angles, and the power spectrum of each, called a diffractogram, shows Thon rings whose radii and orientations encode the aberration coefficients. Fitting the ring structure across all tilt angles yields the full set of estimates. Thon rings are visible only when the specimen scatters electrons uniformly across spatial frequencies [16, 17], as only amorphous materials do.

Both procedures therefore require an amorphous region. This is a persistent operational inconvenience in workflows where the specimen of interest is itself crystalline, such as polycrystalline supports near cryo-EM specimens or battery electrode interphases [16]. Both procedures also require significant expertise and are time-consuming, and as correctors are extended to higher-order terms [18], the number of parameters that must be estimated grows, making manual calibration and monitoring increasingly difficult.

C2: Radiation Damage and Dose Efficiency

Every electron that passes through the specimen deposits energy. In biological and organic materials, this energy breaks chemical bonds through a process known as radiolysis [19]. In crystalline specimens, electrons can displace atoms from their lattice sites. In both cases, the specimen is destroyed by the measurement process itself. This creates a fundamental trade-off: higher electron dose improves signal quality but damages the sample.

Cryo-electron microscopy addresses this by vitrifying specimens and imaging them at cryogenic temperatures, which slows radiation damage enough to allow high-resolution structure determination of biological molecules [20]. In single-particle analysis, many images of the same molecule in different orientations are aligned and averaged, making such structure determination possible even at the low doses that cryogenic imaging requires [10]. The dose limit nevertheless remains one of the most fundamental constraints in electron microscopy, and it is becoming more binding as the field moves toward imaging increasingly beam-sensitive materials. Halide perovskites used in next-generation solar cells, for instance, can suffer measurable structural damage within doses on the order of 100 $e/Å^2$ [21], well below the dose required for conventional high-resolution imaging.

C3: Specimen Positioning and Three-Dimensional Imaging

Accurate specimen positioning is required across many (S)TEM workflows: electron tomography, serial imaging, large-area montaging (tiling many adjacent image fields into a composite), and in situ experiments (where the specimen is modified during acquisition, for example by heating or electrical loading). Electron tomography makes the importance of positioning especially clear. A single electron microscope image is a two-dimensional projection of the specimen, in which all structure along the beam direction is collapsed onto one plane. Tilting the specimen, recording

a projection at each angle, and combining the projections recovers the full three-dimensional structure [22]. This is the same principle as a medical CT scan, in which depth information absent from any single projection is reconstructed from many viewing directions. Electron tomography is constrained from two sides. Geometrically, the maximum tilt angle is limited by the specimen holder, the pole-piece gap, and the specimen geometry itself. The unobserved angles form a missing wedge that degrades resolution along the beam direction and introduces reconstruction artifacts [23]. Mechanically, the projections must all be aligned to a common point of interest, which places severe demands on specimen positioning.

During a tilt series, the specimen stage rotates through a range of angles, yet the point of interest rarely lies exactly on the tilt axis. The three translational actuators that perform fine specimen movement must therefore create a virtual rotation point at the imaging location. These actuators are typically piezoelectric: ceramic elements that expand or contract in response to an applied voltage, with sub-nanometer resolution and no lubricated bearings or gears that would contaminate the ultra-high vacuum [24]. The trade-off is that piezoelectric actuators have well-documented nonlinear pathologies, including hysteresis (displacement depends on the voltage history, so the same voltage can produce different positions) and creep (slow drift after a voltage change) [24]. The encoders that measure displacement are not collocated with the specimen. They sit at the actuator, and the encoder reading deviates from the true point-of-interest position whenever the mechanical linkage bends or drifts. Any positioning error accumulates across the tilt series and degrades the three-dimensional reconstruction [25]. Addressing these sources of error yields improvements across all the workflows mentioned above.

C4: Throughput, Automation, and the Data Challenge

The challenges described above compound in production environments and large-scale research pipelines, where high throughput is a hard requirement. Semiconductor manufacturers inspect wafers at high speed, screening for defects across millions of features. Cryo-EM pipelines process thousands of specimens per year. The data-processing side of these pipelines is now highly automated: tilt-series acquisition is handled by SerialEM [25], and single-particle structure determination by RELION [26] or cryoSPARC [27], with minimal human intervention. The hardware side has not kept pace. The microscope itself must still be calibrated and aligned by human operators before any of this software can run. Manual calibration, alignment, and quality assessment account for a significant fraction of total experiment time. The field is therefore moving toward autonomous operation, where microscopes can calibrate themselves, acquire data, and flag anomalies without constant operator supervision [28, 29].

At the same time, modern detectors are generating data at unprecedented

rates. Direct electron detectors and four-dimensional STEM (4D-STEM) setups can produce gigabytes per second, and the total data volume of a single experiment can reach terabytes. Processing and interpreting data at this scale manually is unfeasible, which has driven the adoption of machine learning for tasks such as denoising, particle picking, and defect detection [30]. Closed-loop operation that spans both the data side and the hardware side of the microscope remains an active challenge.

The challenges described above differ in their physical origins but share two properties that shape what any solution can do. First, every measurement carries a cost: dose limits acquisition in beam-sensitive specimens (C2), tilt acquisitions consume dose and time (C3), and calibration requires trained operators (C1, C4). The volume of real data obtainable in any experiment is therefore hard-bounded. Second, physics-based simulators can generate synthetic observations at low cost, but their output distributions differ from those of real instruments. Hence, two key properties are present across the four challenges:

P1: Limited real data.

P2: Simulation-to-reality gap.

The chapters of this thesis address problems that exhibit P1, P2, or both: real data is physically limited, the available simulator is approximate, or both conditions hold simultaneously. The next section develops the statistical framework common to all of them.

1.3 Learning from Limited and Indirect Measurements

Real-world scientific applications often exhibit the following setting. An observation y carries information about a quantity of interest θ through a forward model $y = f(\theta) + \varepsilon$, where f encodes the physics of the measurement or process and ε is noise. When θ cannot be observed directly but must be inferred from y by inverting this relationship, this constitutes a so-called inverse problem [31]. Bayesian inference is the natural framework for such problems. When treating θ as a random variable, Bayes' rule [32] prescribes how prior knowledge and the data

combine into a posterior distribution:

$$p(\theta \mid y) \propto p(y \mid \theta)\, p(\theta). \tag{1.1}$$

The likelihood $p(y \mid \theta)$ captures how the observations depend on θ through the forward model. The prior $p(\theta)$ encodes what is known about θ before any data is collected. The posterior $p(\theta \mid y)$ is the updated distribution after the data has been taken into account. Both are modeling choices, and these choices determine what can be learned from the data.

Two classical results make the role of these modeling choices more clear. First, for the posterior to concentrate on the true parameter as data accumulates, the mapping $\theta \mapsto p(\cdot \mid \theta)$ must be injective: distinct parameters $\theta_1 \neq \theta_2$ must induce distinct distributions over y. This is the concept of *identifiability*, and without it, no amount of data can resolve which θ generated the observations, regardless of how well-designed the experiments are [33]. Second, even when the model is identifiable, the minimum variance achievable by any unbiased estimator is bounded below by the inverse of the Fisher information [34]: $\mathcal{I}(\theta) = \mathbb{E}[\nabla_\theta \log p(y \mid \theta)\, \nabla_\theta \log p(y \mid \theta)^\top]$. This quantity measures how sensitively the observations discriminate between nearby values of θ.

Together, these results clarify the role of what we will call *structure*: any prior knowledge that constrains the forms the prior and likelihood can take, whether a physical law, a symmetry, a bound on the parameter space, or a dynamical model. Structure determines identifiability, and it determines the Fisher information per observation. In this way, structure controls both *whether* θ can be learned and *how much data is needed* to learn it.

To aid in learning, a simulator can be used to generate synthetic data. Given any θ, the simulator produces synthetic observations at low cost, making it possible to survey the likelihood across the parameter space without collecting real data. The complication is that every simulator is an approximation of reality. When the distribution of simulated observations differs from that of real measurements, the likelihood is misspecified and any resulting posterior is biased. This bias does not vanish with more simulated data, as it reflects a model error, not a statistical one. We call this the *simulation-to-reality gap*. Figure 1.5 shows how structure, real measurements, and simulated data together shape the posterior.

Recent discourse in machine learning has questioned whether structural assumptions retain value as compute and data scale. Sutton's "Bitter Lesson" [35] argues that across the history of AI, general methods that scale with computation have consistently outperformed methods built on human-engineered domain knowledge. The argument is supported by empirical observations across machine learning: in language modeling, loss scales as a power law in model size, data, and compute [36, 37], and in reinforcement learning, AlphaZero outperformed all knowledge-based game programs at chess, shogi, and Go through self-play alone [38].

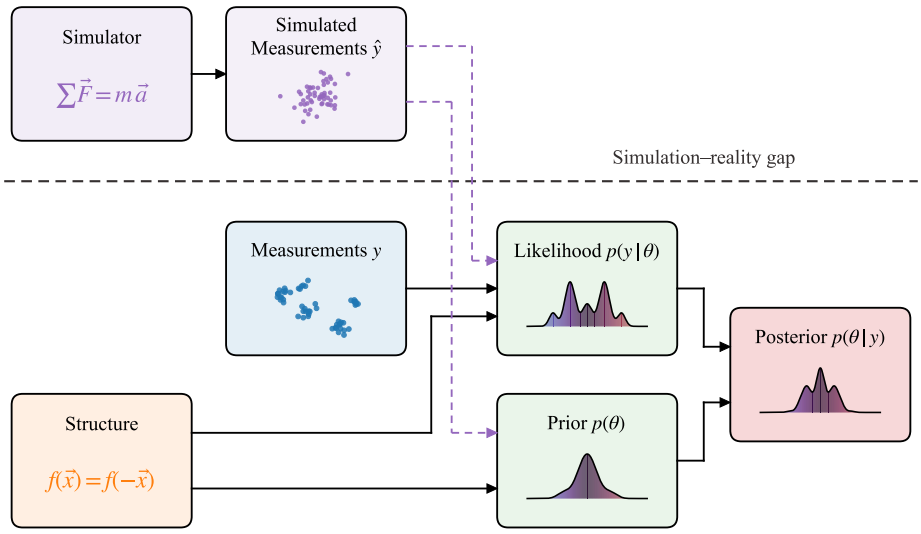


Figure 1.5. Learning from limited and indirect measurements. Structure constrains the forms that the prior and the likelihood can take. Real measurements and simulated data provide evidence through the likelihood; simulated data can also inform the prior. The dashed arrows cross the simulation-to-reality gap: the mismatch between simulated and real data. The posterior combines the prior and the likelihood via Bayes' rule.

If anything required for a problem can be learned from enough data and compute, then structural assumptions are at best a temporary shortcut.

Two recent developments complicate this picture. First, the power-law exponents are small, which means the training scale must be increased significantly to achieve meaningful reductions in loss [36]. Empirical work in vision-language models has confirmed this at the level of individual concepts: zero-shot accuracy on a concept grows log-linearly with that concept's frequency in the pretraining data, so a fixed gain in performance on rare concepts requires exponentially more data [39]. Second, Sutton himself has updated the picture. In a recent essay, he and Silver argue that the human-data era is reaching its limits and that the next phase of AI will rely on agents learning from continuous interaction with the world [40]. Figure 1.6 shows these scaling trends alongside two results from this thesis.

For the problems in this thesis, the Bitter Lesson runs into hard physical limits. The measurement budget is bounded by P1, so the data volumes that scaling requires are simply unavailable. The scaling successes that Sutton cites also share a feature absent here: training and deployment occur in the same environment. In

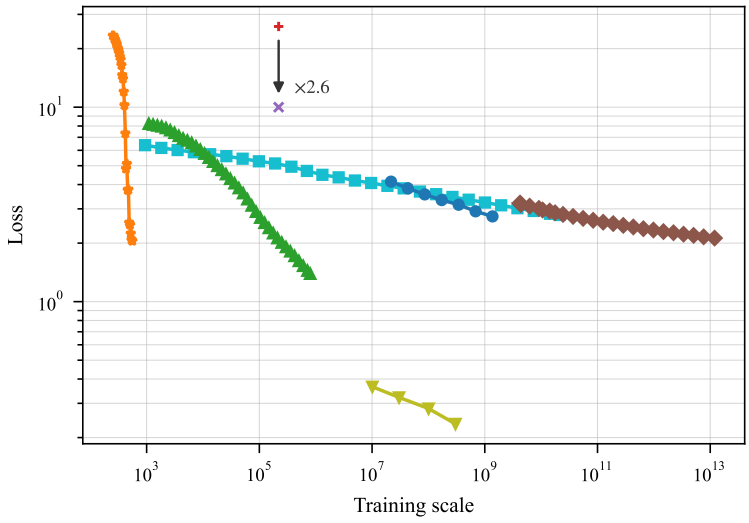


Figure 1.6. Power-law scaling of loss with training scale, shown on log–log axes for published language and vision results together with two data points from this thesis. Each curve keeps its own native metric. The horizontal axis is training scale, given in tokens or images for the data-scaling curves and in GFLOPs for the compute-scaling curves, with compute related to data through $C \approx 6ND$ [36]. The vertical axis is the loss or error reported for each curve. For language modeling, the loss follows a power law both in dataset size ([36]) and in compute ([36]), and a compute-optimal fit gives the same trend ([37]). The same power-law behavior appears in vision, for ImageNet classification ([41]) and for JFT segmentation ([42]). The exponents are small, ranging from $\alpha \approx 0.3$ for image classification down to $\alpha \approx 0.1$ for language modeling, so halving the loss takes roughly $2^{1/\alpha}$ times more data, from about ten to about a thousand times, and each further halving demands the same multiplicative increase. Structure in the training data can beat this scaling. Pruning redundant examples turns the power law into a faster decay ([43]), so a well-chosen subset reaches an error that brute-force scaling reaches only with far more data. The two thesis points follow the same principle at a fixed training-set size of over 200 000 simulated Ronchigrams. Replacing the scalar Bayesian-optimization approach (Chapter 2, +) with the more structured VAE-EM methodology (Chapter 3, ×) lowers the median aberration estimation error by 2.6× without any additional training data.

electron microscopy the training environment is a simulator and the deployment

environment is a real instrument.[2] P2 means that scaling the simulated dataset cannot eliminate the distributional mismatch. On top of this, many of the problems considered in this thesis are inverse problems, and so identifiability depends on the structural assumptions one is willing to make. For some, no amount of data resolves θ unless a prior constrains the hypothesis space.

For this reason, the central question of this thesis is: *how can structural knowledge enable learning from limited and indirect measurements?* The next section describes the contributions and the organization of the thesis.

1.4 Thesis Outline

The eight content chapters of this thesis are organized into three parts, progressing from application-specific methods for electron microscopy to general methodology for data-efficient machine learning. Figure 1.7 at the end of this section provides a graphical overview.

Part I: Machine Learning for STEM Calibration

Hardware aberration correctors, introduced in the late 1990s, made sub-ångström STEM imaging possible [7, 8]. However, the correctors also lead to a tremendous growth of the number of calibration parameters, and recalibrating them still requires expert interpretation of Ronchigrams. Two paths have emerged for automating this. The first is physics-based, with feature-engineered methods that exploit the optical model directly [44–46], used today in commercial tools such as CEOS [47]. These methods are interpretable but limited by the accuracy of the underlying model and the difficulty of capturing day-to-day variation in real microscopes. The second is data-driven, applying deep learning or Bayesian optimization to features of the Ronchigram [48–52]. Reviews of deep learning in electron microscopy identify this direction as the natural complement to the automated software pipelines discussed in C4 [29, 30, 53]. Both chapters in Part I contribute to the data-driven path and address the aberration estimation aspect of C1, reducing reliance on expert operators (C4).

Chapter 2 establishes the data-driven baseline for STEM calibration. Aberration states drift frequently, and recalibration by hand requires expert interpretation of Ronchigrams. The pipeline separates calibration into two steps. First, a ResNet18 convolutional neural network trained on a large dataset of simulated Ronchigrams assigns a scalar quality metric to each image, reflecting how far the instrument is from its calibrated state. Second, a Bayesian optimization loop uses a Gaussian

[2] Real EM data is used in each chapter, but only in limited amounts and only to fit the parameters of highly constrained models or to validate performance, not to train general methods from scratch.

process model of this quality surface and a problem-specific acquisition function to select corrector adjustments iteratively; the acquisition function is designed around the geometry of the calibration problem and outperforms standard alternatives. Experiments on a real STEM show that the method consistently calibrates defocus and two-fold astigmatism to within 15 nm and 20 nm, respectively, using roughly twenty Ronchigrams acquired in under fifteen seconds. A scalar quality metric, however, compresses the Ronchigram into a single number. As the number of aberration types grows, this compression loses resolution.

Chapter 3 addresses the two main limitations of the previous chapter: the scalar representation and the simulation-to-reality gap. A variational autoencoder (VAE), trained on simulated Ronchigrams, learns a multi-dimensional latent representation that retains more information about the aberration state. The key challenge is that the VAE is trained in simulation, so its latent mapping is biased relative to real instruments. An expectation-maximization algorithm addresses this by jointly estimating both the aberration state and a Gaussian process model of the remaining mismatch between the simulated and real domains — that is, it learns the simulation-to-reality gap directly from a small number of real measurements. The joint estimation is otherwise ill-posed, but the even-symmetry of the Ronchigram power spectrum (a consequence of Fourier optics) constrains the GP kernel and restores a unique solution. On a real STEM, the method more than halves the aberration estimation error compared to the previous approach and converges within tens of seconds.

Part II: Image-Based Calibration and Control of TEM

Both chapters in Part II address front-end TEM tasks where automation has lagged behind the data-side software pipelines introduced in Section 1.2. Chapter 4 addresses aberration calibration in TEM (C1), and Chapter 5 addresses specimen positioning during a tilt series (C3). They share a common methodological idea: the microscope's detector, normally used only to record the final image, is repurposed as a measurement sensor for the front-end task. Together the two chapters target the throughput bottleneck (C4) that is sharpest in electron tomography, where fast aberration calibration and precision specimen positioning are jointly required for a usable tilt series. The methods themselves are not specific to tomography.

Chapter 4 introduces a TEM aberration estimation method based on beam tilts. Tilting the electron beam in the presence of aberrations causes a lateral shift in the image whose direction and magnitude encode the current aberration state. The tilt-shift principle has been used for aberration estimation before [54, 55], but acquisition and processing times prevented earlier methods from matching the Zemlin tableau in practice and from tracking aberration drift in real time. This chapter combines fast GPU-based image-shift extraction [56] with a sensor-

scheduling formulation. A Kalman filter estimates the aberration coefficients and their drift jointly, and the question becomes which sequence of beam tilts extracts the most information given the measurement budget. The tilt schedule is optimized offline by minimizing the trace of the predicted estimation covariance (A-optimality), using a receding-horizon gradient method with multiple initializations. A supplementary expectation-maximization step estimates specimen-dependent noise parameters, adapting the schedule to the specific specimen being imaged. The model predicts that optimized schedules reach the estimation accuracy of uninformed patterns in roughly 42% fewer tilts, and experiments on a real TEM reproduce the predicted estimation covariances. The method also generalizes to polycrystalline specimens, where the Zemlin tableau cannot be applied (Section 1.2).

Chapter 5 addresses precision specimen positioning in electron tomography, where the specimen must remain at the same point of interest throughout a full tilt series (C3). The four-degree-of-freedom motion stage used for positioning suffers from several sources of error: hysteresis in the piezo stepper actuators, repeatable stepping errors from mechanical misalignments, and a non-collocated sensing arrangement where encoders sit at the actuators rather than at the specimen. No dedicated sensor observes the actual specimen position. The baseline approach used in this chapter follows van Meer et al. [57], which developed data-driven hysteresis compensation and commutation-angle-domain iterative learning control for piezo-stepper actuators on a laboratory setup. This chapter extends that work to an operational TEM and provides the missing specimen-position sensor by using the microscope images. The proposed framework addresses the error sources jointly. The system is first linearized by inverting the identified parametric hysteresis model. Repeatable stepping errors are canceled by iterative learning control in the commutation-angle domain, extended from one to three translational actuators. The missing point-of-interest measurement comes from the microscope images themselves. Cross-correlation-based tracking of consecutive images provides an in-plane position estimate. This estimate can then be used to provide a learned sensor deviation that corrects encoder estimates for quasi-static bending at the contact points, which allows the ILC framework to be applied directly to a POI position estimate rather than to the encoder readings. The complete framework is validated on a four-degree-of-freedom lab setup and on an operational electron microscope and achieves nanometer-level positioning accuracy, reducing the point-of-interest tracking error by roughly an order of magnitude relative to the baseline on both platforms.

Part III: Data-Efficient Machine Learning

Section 1.3 argued that in problems where data is bounded, structural assumptions are how a method stays useful. The four chapters in Part III put this argument

into practice with general methods that are independent of the electron microscopy application. They sit in three research areas where structural assumptions have a long history of making inference tractable. Representation learning has long sought structural priors that match the geometry of the data, beyond what a Gaussian prior can express [58]. Spatiotemporal estimation has historically combined kernel methods with dynamical models to reduce continuous-domain estimation problems to tractable finite-dimensional ones [59]. Reinforcement learning faces an explicit sample-complexity problem, and surveys on Bayesian and model-based RL [60, 61] catalog a wide range of structural assumptions that extract usable policies from a small number of interactions. Part III contributes one chapter to representation learning, one to spatiotemporal estimation, and two to reinforcement learning.

Chapter 6 addresses a structural limitation of the VAE in Chapter 3: the standard Gaussian prior imposes a Euclidean topology on the latent space, which is a mismatch when the data lies on a manifold with different structure. For Ronchigrams, the optical symmetry already exploited in Chapter 3 suggests a non-Euclidean latent topology, and many other datasets in engineering and science lie on tori, cylinders, or similar manifolds. The chapter introduces a constructive procedure for any manifold that can be expressed as a product of elementary factors (circles, intervals, and lines) or as a quotient of such a product by a finite group. Factorized distributions over these factors yield product topologies with closed-form, decoupled KL divergences. For quotient manifolds, a group-invariant feature map serves as the decoder input, ensuring that identified points produce identical outputs. Anchor constraints fix the coordinate orientation or create topological holes. On three synthetic datasets and two modified MNIST datasets, geodesic stress relative to a Gaussian VAE baseline decreases by 77% for a cylindrical latent space, by 41% for a toroidal one, and by 74% for a Möbius strip.

Chapter 7 provides a unified estimation framework for functions that both vary over a continuous spatial domain and evolve over time. Standard Kalman filtering handles finite-dimensional dynamics but not spatial fields; Gaussian process regression handles spatial fields but not temporal evolution. The Dynamic Gaussian Process (DGP) addresses both by modeling the function as a Gaussian process whose state evolves through an integro-difference equation. The posterior remains a Gaussian process with closed-form mean and covariance updates after each evolution step and each set of observations. When the spatial and temporal kernels are separable, the infinite-dimensional problem reduces exactly to a finite-dimensional Kalman filter on basis function coefficients. This reduction unifies both frameworks under a single formulation. A stability and approximation error analysis decomposes the functional estimation error into three contributions: a noise-limited term, a basis-leakage term, and an out-of-subspace residual. The latter two vanish as the number of basis functions grows, while the noise-limited term converges to the posterior uncertainty of the exact infinite-dimensional estimator.

The framework is extended to vector-valued states, enabling the treatment of higher-order partial differential equations via state augmentation, and demonstrated on the heat equation and on the wave equation.

Chapter 8 addresses sample efficiency in reinforcement learning through structural assumptions on the Q-function. Standard Q-learning requires visiting each state-action pair many times before reliable value estimates emerge, because no structure is imposed on the value function. The approach here restricts the Q-function to a quadratic form in selected state and control basis functions, which is appropriate for problems near a known base policy. The learning objective is to find the quadratic parameters that minimize the ℓ_1-norm of the Bellman residuals over a collected dataset. This problem is not convex, but two tractable convex approaches are derived. LMI-QL applies a semidefinite programming relaxation involving linear matrix inequalities. LMI-QLi instead fixes a subset of the parameters at each step and solves a semidefinite program for the rest, iterating until convergence. Both are off-policy methods, so the data can be collected in any way. On a nonlinear pendulum benchmark, both methods learn an accurate value function from fewer interactions than the least-squares policy iteration baseline.

Chapter 9 addresses slow learning from uninformed exploration in reinforcement learning. When an agent has no model of the environment, it must explore randomly, visiting many state-action pairs that are provably suboptimal. The key observation is that in many problems a model set can be specified from prior knowledge: a set of transition models that contains the true dynamics. Optimizing over this model set yields upper and lower bounds on the Q-function. Any action whose upper bound falls below the lower bound of the best currently known action is provably suboptimal and can be pruned, directing exploration toward the remaining candidates. A data-regularized version of the model set optimization incorporates observed transitions, which allow us to close the gap between the model and the true system as the data accumulates. The regularization also ensures convergence of the exploring policy to the optimal one. For the bounded-parameter MDP (BMDP) framework, the model set optimization is convex, yielding the BUMEX algorithm (Bounded Uncertainty Model-based Exploration) with finite-time convergence guarantees under mild assumptions. On standard benchmark environments including Frozen Lake, Cartpole, and Taxi, BUMEX converges to a good policy using substantially fewer interactions than ϵ-greedy and UCB-based baselines.

Chapter 10 concludes the thesis with a summary of the main findings and directions for future research.

Publications Underlying Each Chapter

The chapters of this thesis are based on the peer-reviewed publications listed below. A complete list of publications, including co-authored work that does not correspond to a chapter, appears at the end of this thesis.

Chapter 2: J.S. van Hulst, R.A.C. van Zuijlen, N. Javaheri, M. Diephuis, D.J. Antunes, and W.P.M.H. Heemels, "Calibration of Electron Microscopes Through Deep Learning and Bayesian Optimization," *IEEE Access*, vol. 13, pp. 137 291–137 302, 2025.

Chapter 3: J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Bridging the Simulation-to-Reality Gap in Electron Microscope Calibration via VAE-EM Estimation," under review, arXiv:2603.16549, 2026.

Chapter 4: J.S. van Hulst, E.M. Franken, B.J. Janssen, W.P.M.H. Heemels, and D.J. Antunes, "Tilt-Based Aberration Estimation in Transmission Electron Microscopy," *IFAC Mechatronics*, vol. 117, p. 103529, 2026.

Chapter 5: J.S. van Hulst, A.M.C. de Peffer, D. Herceg, E.M. Franken, E. Verschueren, W.P.M.H. Heemels, and D.J. Antunes, "Precision Specimen Positioning in Electron Microscopy through Hysteresis Compensation, Iterative Learning, and Vision-Based Sensing," under review, arXiv:2608.03669, 2026.

Chapter 6: J.S. van Hulst, J.M. Tomczak, W.P.M.H. Heemels, and D.J. Antunes, "Constructing VAE Latent Spaces with Prescribed Topology," under review, arXiv:2606.07058, 2026.

Chapter 7: J.S. van Hulst, R.A.C. van Zuijlen, D.J. Antunes, and W.P.M.H. Heemels, "Estimation of Dynamic Gaussian Processes," in *Proc. 62nd IEEE Conference on Decision and Control (CDC)*, Singapore, 2023, pp. 3206–3211.
J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Estimating Evolving Functions with Dynamic Gaussian Processes," under review, arXiv:2606.06705, 2026.

Chapter 8: J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Data-Efficient Quadratic Q-Learning Using LMIs," in *Proc. 63rd IEEE Conference on Decision and Control (CDC)*, Milan, 2024, pp. 1161–1166.

Chapter 9: J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Smart Exploration in Reinforcement Learning Using Bounded Uncertainty Models," in *Proc. 64th IEEE Conference on Decision and Control (CDC)*, Rio de Janeiro, 2025, pp. 5132–5138.

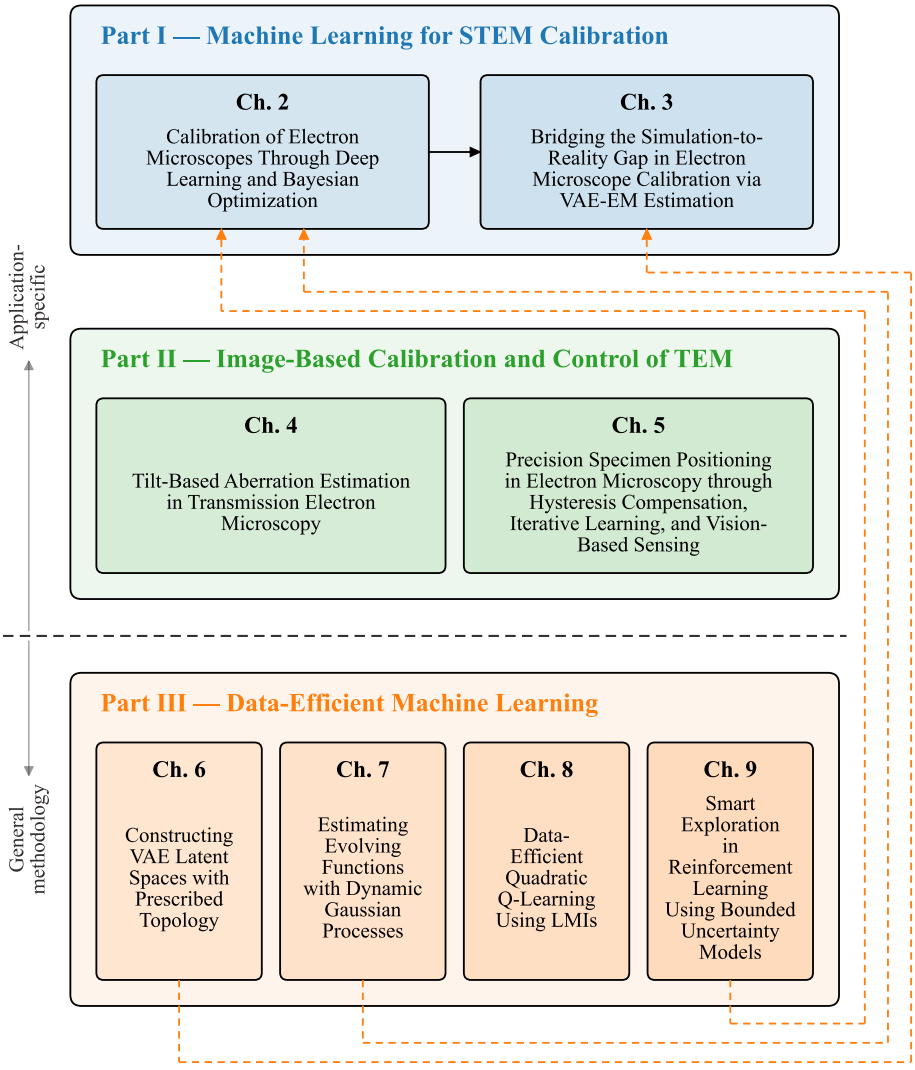


Figure 1.7. Graphical overview of the thesis. Solid arrows indicate direct methodological progression. Dashed arrows indicate cross-part connections: the Dynamic Gaussian Process (Chapter 7) extends the Gaussian process surrogate at the core of Chapter 2 to the spatiotemporal setting; the topology-aware VAE (Chapter 6) extends the architecture of Chapter 3 with the same optical symmetry; and the exploration framework (Chapter 9) adapts the acquisition function heuristics of Chapter 2 to reinforcement learning.

I

Machine Learning for STEM Calibration

2

Calibration of Electron Microscopes Through Deep Learning and Bayesian Optimization

Abstract - Electron microscopes can create images at atomic resolution, enabling key developments across many scientific fields. To obtain high-resolution images, the inherent imperfections in the electromagnetic lenses have to be overcome by frequent and extensive calibration. Currently, this calibration is typically performed or monitored by experts, making it expensive and time-consuming. To resolve this, this chapter proposes a data-driven solution for automated, image-based calibration of a Scanning Transmission Electron Microscope (STEM). The method uses deep learning to interpret the high-dimensional image, also called the Ronchigram, produced by the STEM, providing a quality index that quantifies the distance of the current image from its calibrated state. Specifically, we perform transfer learning on an adapted ResNet18 architecture using a large dataset of simulated Ronchigrams. Afterward, Gaussian process regression models how calibration settings affect this quality index from limited samples. Subsequently, candidate calibration settings are selected online using Bayesian optimization with a tailored acquisition function that, like entropy search, selects settings to reduce the

This chapter is based on van Hulst et al. [62].

uncertainty about the calibrated state. When applied to the STEM, the proposed method consistently calibrates defocus and two-fold astigmatism in just a few steps. Compared to existing automated calibration methods, the proposed method can consistently achieve similar performance in less time. For this reason, our approach potentially reduces expert involvement and improves the overall throughput of the STEM. Moreover, the data-driven nature of the proposed method suggests it could be adapted to calibrate additional types of imperfections, positioning it as a flexible tool for enhancing STEM performance, provided that representative data can be generated from the simulator.

2.1 Introduction

Electron microscopes (EMs) are able to create images at atomic resolution. Such images are used in, for instance, state-of-the-art materials science, life science, and the semiconductor industry. The high magnification is enabled by firing a focused beam of electrons onto a specimen and measuring the electrons after they have interacted with the specimen. The quality of the EM image is directly related to the quality of the electron beam focus, which is achieved by electromagnetic lenses that inherently suffer from optical aberrations.

Aberrations are imperfections in the focus of the lenses that cause the resulting image to be blurred. One type of aberration is called astigmatism, in which the focus point of the electron beam is dependent on the angular coordinate of the beam's entry point on the lens. The unaligned focus points result in poor image quality [63]. For this reason, stigmators were introduced in EM, a hardware component that can actively compensate astigmatism by manipulating the magnetic field [64, Chapter 4]. Another type of aberration is called spherical aberration, where the refraction of the electron beam is more or less severe at the center of the lens compared to the edges. Spherical aberration can also be compensated by the introduction of additional hardware for the EM [6].

The introduction of these hardware components has dramatically increased the attainable resolution in EMs [65]. However, these components have also increased the number of settings that need to be calibrated in order to get high-quality images. The performance of the EM in terms of image quality is extremely sensitive to these calibration settings. Additionally, the optimal settings are dependent on many factors, such as the magnetic composition of the specimen and the EM surroundings [66, 67]. The problem is exacerbated by the fact that aberrations typically drift from day to day, with some beginning to drift within minutes, and that typical specimens can only take a limited number of electrons before significant damage occurs.

Currently, the calibration of the STEM is performed by an expert microscope operator. Although there are multiple methods for calibrating STEM that aim at automatic calibration, most still require expert intervention. One of the calibration modes used in STEM involves Ronchigram images. A Ronchigram is an electron diffraction pattern from a convergent beam diffracted by a sample, typically amorphous materials. This pattern encodes information from the incoming electron beam and its aberrations [68]. The image is formed on a pixelated camera at the end of the TEM column. Note that this image is an extremely high-dimensional signal, as the number of pixels in a single Ronchigram is well over 10 000. On top of that, different aberrations can result in Ronchigrams that cannot be told apart, such as in the case of underfocus and overfocus, which both appear as a zoomed-out image.

The frequency and duration of recalibration severely limit the number of high-quality images that can be produced within a given time frame [69]. Moreover, image quality often suffers due to suboptimal settings [70]. High throughput and image quality are critical in industrial settings, as they directly impact the efficiency and scope of materials science, life science, and semiconductor research. For this reason, an automated calibration method for electron microscopes holds significant value.

We propose to tackle the calibration problem using a two-step approach:

1) **Interpreting the Ronchigrams**: This step leverages deep learning trained on a large dataset of simulated Ronchigrams. A scalar quality value is assigned to each Ronchigram that represents the distance from calibration. The neural network architecture is based on ResNet18 [71], modified with a fully connected layer to produce scalar outputs.

2) **Decision-making based on the interpretations**: The outputs from step 1) are used in a Bayesian optimization framework [72], which employs a Gaussian process model and a problem-specific acquisition function that explicitly minimizes the expected calibration error.

This fully data-driven approach enables rapid and consistent tuning of lower-order aberrations, such as defocus and two-fold astigmatism. Due to its data-driven foundation, the methodology is likely to be generalizable to other types of aberrations, a prospect further explored in Section 2.6.

This is not the first work to attempt automatic aberration correction in EM, and significant progress has been made in this field in recent years. Traditional EM calibration methods usually rely on hand-crafted features obtained from (a sequence of) Ronchigrams, which are then used to estimate the aberrations. In Vulović et al. [44], a direct aberration estimation method is proposed that uses the power spectral density of the Ronchigram. In Rudnaya [45, Chapter 4], the Fourier transform is applied to the Ronchigram to estimate the aberrations, followed by a derivative-based calibration of the EM. In Rudnaya et al. [73], a fully automated gradient-based calibration method is proposed for astigmatism and defocus based on image variance. These methods presented the early beginnings of automated STEM calibration, but have since been surpassed in terms of speed and accuracy. The most widely used method for STEM calibration in industry nowadays is by the company CEOS, which can typically calibrate lower-order aberrations up to 10 nm each [47]. The STEM manufacturer Thermo Fisher Scientific supplies their microscopes with OptiSTEM(+), which can achieve similar performance [46]. Both of these methods rely on physics-based insights from the full STEM images, which take longer to acquire than Ronchigrams. The physics-based nature suggests that these methods are prone to undermodeling, which is evident in practice since expert preparation or intervention is often necessary.

More recent works have investigated the use of machine learning to automate the calibration process. For instance, Bertoni et al. [48] proposes the use of artificial neural networks to estimate aberrations based on Ronchigrams. In Schubert et al. [49], the DeepFocus method is proposed, which uses deep learning to estimate aberrations in the scanning electron microscope. Recent results have even shown that deep learning can interpret certain aspects of the Ronchigram with greater accuracy than experienced microscopists [52]. Another key technology that has been introduced in EM calibration is Bayesian optimization. Our previous work introduces Bayesian optimization based on hand-crafted (Fourier) features of Ronchigrams [74]. In Ma et al. [50], an alignment method for STEM and ptychography is proposed using Bayesian optimization of the beam emittance, a physical property of the Ronchigram. The work Pattison et al. [51] also implements a version of Bayesian optimization for the calibration of the STEM, but specializes to low-valued aberrations.

Although other works have addressed the automatic calibration of electron microscopes, this is the first to adopt a fully data-driven approach with state-of-the-art calibration speed and accuracy. To achieve this, we propose a model-free adapted image-recognition neural network (ResNet) to learn the map from Ronchigrams to distance to a calibrated state in a supervised manner. Moreover, we propose a tailored acquisition function for Bayesian optimization that treats calibration as a pure-exploration problem. In the spirit of entropy search, it selects settings that reduce the uncertainty about the calibrated state rather than greedily minimizing the estimated aberration norm. It thereby speeds up calibration compared to standard acquisition functions and provides a natural stopping criterion. These features ensure consistently high calibration performance and a strong potential for generalization to various types of aberrations, including higher-order aberrations.

2.2 Background and Problem Formulation

In this section, we formulate the problem addressed in this chapter. First, background on electron microscopy and the calibration process is provided. Afterward, the problem is explicitly stated.

2.2.1 Scanning Transmission Electron Microscopy

The scale at which traditional light microscopes can image is fundamentally limited by the wavelength of visible light. By using electrons instead of photons, which have a wavelength about 100 000 times as small, electron microscopes are able to achieve superior magnification.

There are different types of EM, each with its own specific advantages and disadvantages. Transmission Electron Microscopes (TEM) operate similarly to

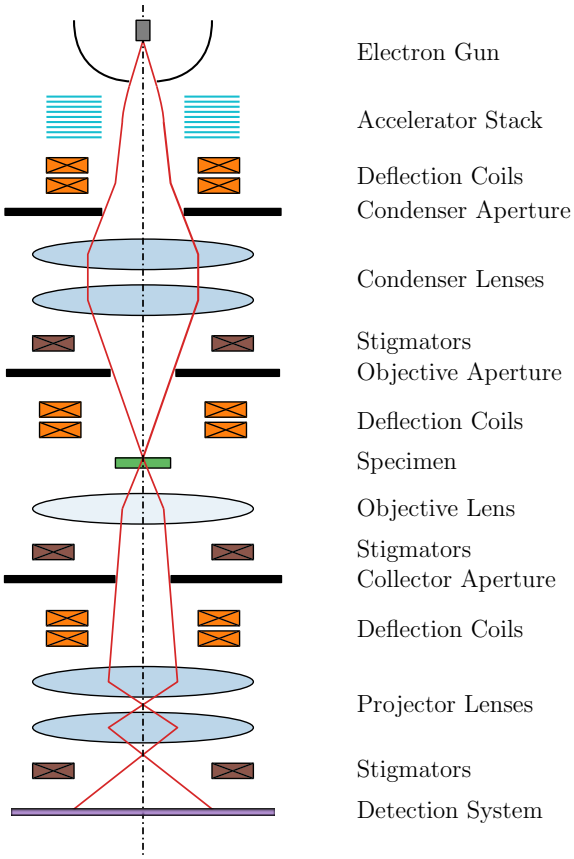
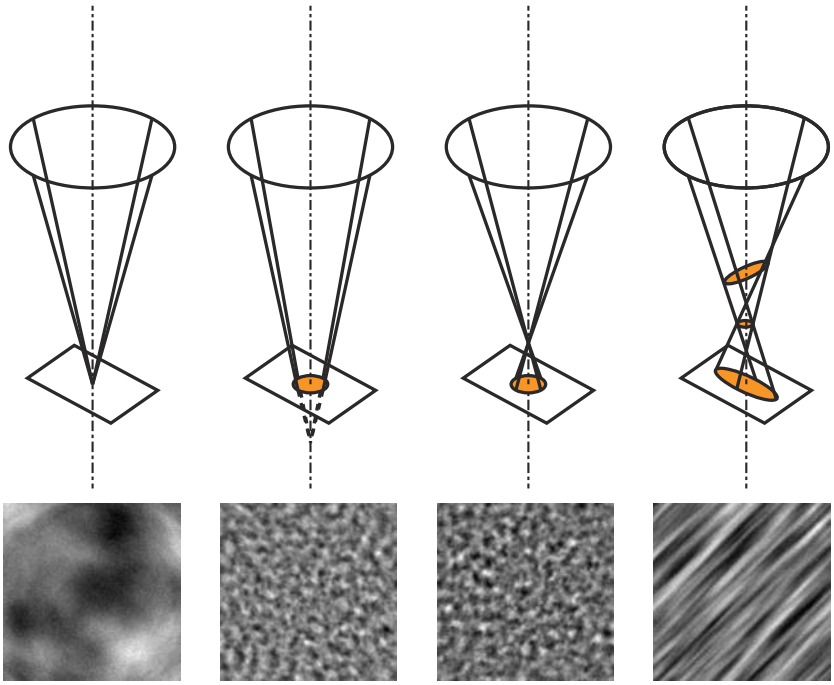


Figure 2.1. Schematic representation of STEM. An electron gun fires a beam of electrons into a stack of lenses and apertures. Between the lenses, there are stigmators and deflection coils which allow for active compensation of the lens aberrations. The transmitted electrons are collected after interaction with the specimen in a detection system.

light microscopes in the sense that the image is produced by detecting the electrons that have been transmitted through a specimen. Scanning Electron Microscopes (SEM) instead image by focusing the electron beam on the surface of the specimen and detecting the secondary and backscattered electrons. This technique allows for scanning along the surface of the specimen, but it has a lower resolution when compared to TEM. The Scanning Transmission Electron Microscope (STEM) combines both of these techniques. It creates an observation from transmitted electrons at high resolution, but with the possibility of scanning along the specimen's



(a) Calibrated **(b)** Underfocus **(c)** Overfocus **(d)** Astigmatism

Figure 2.2. Defocus and two-fold astigmatism aberrations visualized along with the resulting Ronchigrams, inspired by [75]. Each panel shows the ray diagram above and the corresponding Ronchigram below. Panel **(d)** combines two-fold astigmatism with overfocus.

surface. This enables analysis of larger parts of the specimen. A diagram of a typical STEM is shown in Figure 2.1.

2.2.2 Problem Formulation

Due to the inherent limitations of electromagnetic lenses, the focus of the electron beam is prone to imperfections [3]. Figure 2.2 illustrates how these imperfections can result in failure to focus onto a single point, resulting in poor images. The figure illustrates both defocus (C1) and two-fold astigmatism (A1), which are referred to as low-order aberrations and are the most rapidly changing aberrations, thus requiring the most frequent recalibration. These parameters are therefore the calibration variables considered in this chapter. Defocus occurs when the lens bends the electrons either too much or too little, resulting in the beam converging

before or after the desired position, respectively. The figure also illustrates two-fold astigmatism, in which electrons that travel in perpendicular planes have different focus points. In combination with defocus, the electron beam is stretched, which results in an elliptical projection instead of a circular one. Note that even with perfect focus, the electron beam cannot converge onto a single point in the presence of astigmatism.

By providing the lenses and stigmators with the correct voltages, it is possible to compensate defocus and astigmatism, resulting in sharp images even with imperfect lenses. The way these voltages are tuned and selected is by interpreting the Ronchigram [68]. This Ronchigram is an interference pattern created by the transmitted electron wavefronts by capturing the transmitted electrons in a STEM for a known amorphous material. Under no aberrations, these transmitted electrons generally exhibit no structure or patterns. Conversely, any structure or patterns in the Ronchigram can be related to the aberrations. For instance, underfocus and overfocus result in a zoomed-out Ronchigram, see Figure 2.2. Interestingly, there are more cases where different aberrations result in Ronchigrams that cannot be told apart. Two-fold astigmatism shows up in the Ronchigram as a stretched structureless pattern in one direction, as shown in Figure 2.2.

The calibration problem tackled in this chapter can be summarized as containing the following features:

- The calibration of the STEM requires bringing the aberrations to zero by choosing the appropriate tuning knob values.
- The calibration is performed based on the high-dimensional Ronchigram image, even though the aberrations are of much lower dimensions, and therefore the differences between Ronchigrams have a low-dimensional structure.
- Aberrations can often not be uniquely inferred from a single Ronchigram.

In the next section, we formalize the calibration problem and objective using the definitions and equations that follow.

2.2.3 Mathematical Problem Setup

Let the microscope state $x(t) \in \mathbb{R}^n$ represent the current aberration levels at the discrete time index $t \in \mathbb{N}$, with $x(t) = 0$ corresponding to a fully calibrated state. In this sense, the distance from the origin can be viewed as the quality of the microscope calibration. The state $x(t)$ cannot be measured directly, complicating the calibration process. Instead, we observe the system output, the Ronchigram, as $y(t) \in \mathbb{R}^p$. User-controlled inputs are denoted $u(t) \in \mathcal{U} \subset \mathbb{R}^m$. These inputs are translated into currents for the electromagnetic lenses and stigmators in the EM, changing the alignment of the electron beam, resulting in a change in the current

aberration level. Note that for each element of the state vector, there is exactly one corresponding element of the input vector dedicated to its calibration, hence $n = m$. The calibration objective is to bring $x(t)$ to the origin by finding a sequence of inputs based on the observed outputs. It is assumed that the aberrations remain the same in the absence of changes in the inputs. We can capture these dependencies in the following equations:

$$\begin{aligned} x(t) &= \xi + u(t), \\ y(t) &= g(x(t)) + \eta(t), \end{aligned} \tag{2.1}$$

where $\xi \in \Xi \subset \mathbb{R}^n$ represents the aberration level in the absence of compensating inputs, which is unknown and typically different every time the calibration problem is performed; $\eta(t) \in \mathbb{R}^p$ is a random variable with zero mean and an unknown distribution, which represents the measurement noise at time $t \in \mathbb{N}$, and $g : \mathbb{R}^n \to \mathbb{R}^p$ describes the generation of every pixel in the Ronchigram based on the current aberrations.

In general, the function g is unknown and extremely complex. Note that the number of pixels is much larger than the number of aberration types to be corrected, i.e., $p \gg n$. Additionally, the map g cannot be inverted in practice, since distinct states $x \neq x'$ can produce Ronchigrams that cannot be told apart. Consider, for example, overfocus and underfocus, where the two middle Ronchigrams in Figure 2.2 look alike, even though the states (under versus overfocus) are completely different and require different inputs to achieve optimal calibration. This makes it impossible to find the correct inputs using a single observation with certainty. In fact, a sequence of multiple input-output pairs is needed to find the input $u(t)$ that brings $x(t)$ to the origin. Additionally, the presence of measurement noise on the outputs also implies that a history of outputs $y(t)$ is required in order to successfully calibrate with high certainty, akin to a Bayes filter [76].

The calibration procedure is considered successful when we are sufficiently confident that we have reached the calibrated state $x(t+1) = 0$ within some tolerance $\tau \in \mathbb{R}_{\geq 0}$, i.e.,

$$\text{Prob}\left[\|\xi + u(t+1)\|_2 \leq \tau \mid \mathcal{D}(t)\right] \geq \epsilon, \tag{2.2}$$

where $\epsilon \in (0, 1)$ represents the required confidence. The set $\mathcal{D}(t) := \{(y(i), u(i))\}_{i=0}^{t} \in (\mathbb{R}^p \times \mathcal{U})^{t+1}$ denotes the data gathered from previous attempts, which can be used to select the input $u(t+1)$.

2.3 Methods

This section details the approach proposed in this chapter to tackle the STEM calibration problem. Conceptually, the proposed method is divided into two steps:

1. The interpretation of the Ronchigrams, which is performed through the use of deep learning based on a large dataset of simulated Ronchigrams in this research;

2. The decision-making based on the interpretations obtained in 1), which is performed through Bayesian optimization in this research.

This research utilizes both a high-fidelity STEM simulator and an experimental STEM setup. The simulator is based on first-principles wave optics, specifically modeling the specimen-electron interaction with the multi-slice method. Due to the nature of simulations, we have access to exact aberration values for simulated Ronchigrams and can generate large datasets quickly. In contrast, acquiring Ronchigrams in the experimental setup is more time- and resource-intensive, resulting in limited datasets unsuited for training data-driven models. However, experimental datasets are valuable for validating our methods, as the simulator is prone to undermodeling, and the experimental STEM setup is subject to machine-to-machine and day-to-day variations not captured by the simulation. Due to the proprietary nature of the simulator, further details cannot be disclosed. For the rest of this chapter, we will denote the set of simulated Ronchigrams by $\mathcal{D}_S$, while three experimental datasets gathered from an actual STEM setup on different days will be denoted by $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$.

Figure 2.3 shows the complete pipeline of the proposed method. The interpretation is performed in blocks **(b)** and **(d)** and is presented in detail in Section 2.3.1. The decision-making is performed in **(e)** and is detailed in Sections 2.3.2 and 2.3.3.

2.3.1 Deep Learning

The primary objective of employing deep learning is to reduce the dimensionality of high-dimensional Ronchigram images while preserving the critical information needed for decision-making. We use a supervised learning approach to predict the 2-norm of the aberration vector, which is discussed later in this section. To achieve this, we adapt the ResNet18 architecture, a well-known image classification network [71], to produce a scalar output. Leveraging its pre-trained image recognition capabilities, we modify the network by adding a fully connected layer at the end that generates the scalar prediction. Similar ideas are discussed in the survey on transfer learning for medical image analysis performed in [77].

To enhance the performance of supervised learning, we apply several preprocessing steps. First, we apply a Hanning window to reduce spectral leakage. This operation is followed by a 2D Fourier transform to convert the Ronchigrams into the frequency domain, where differences in stretching and magnification are more pronounced; see Figures 2.4 and 2.5. Next, we compute the squared magnitudes of the Fourier-transformed images, which represent the power distribution. Lastly,

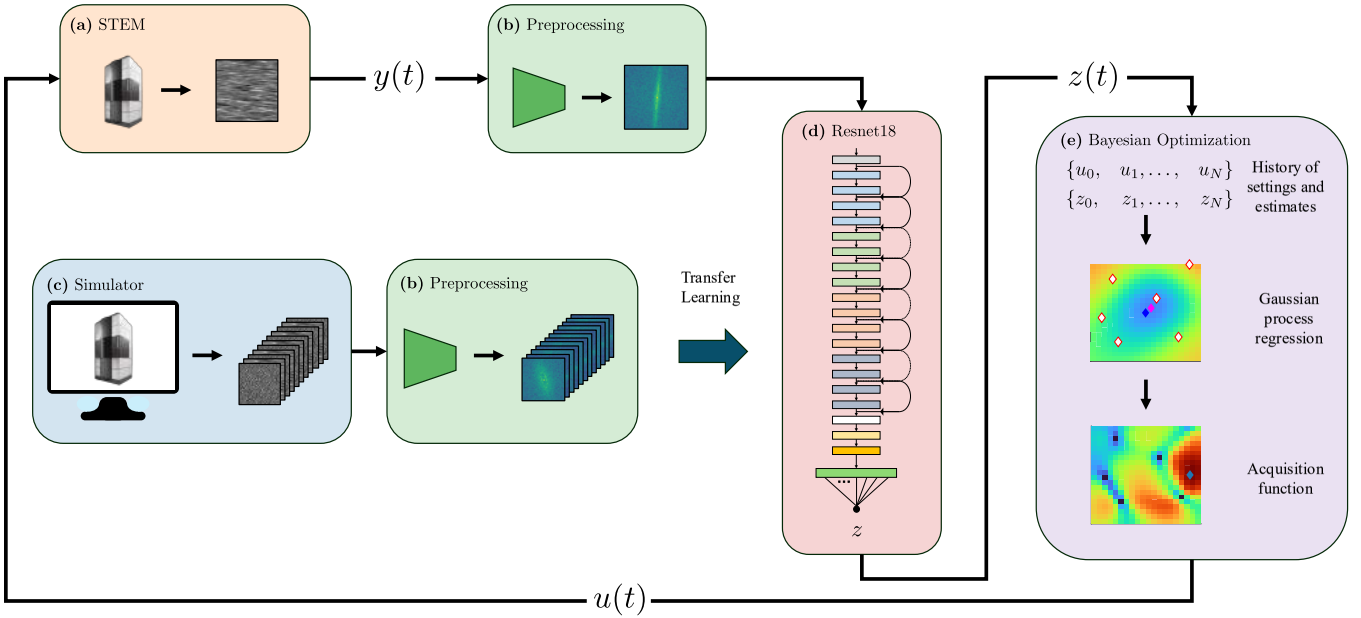


Figure 2.3. **(a)** Experimental STEM setup, which outputs Ronchigrams in real time based on calibration settings. **(b)** Preprocessing of Ronchigram images, which includes taking the 2D Fourier transform preceded by a Hann windowing operation. **(c)** Simulator generating large dataset of Ronchigram images with various known aberrations. **(d)** Adapted ResNet18 architecture, where the classification layer is followed up by a fully connected scalar output layer tasked with estimating the aberration 2-norm of a Ronchigram. **(e)** Bayesian Optimization algorithm which takes in all Ronchigrams and calibration settings used thus far and proposes new calibration settings based on Gaussian process estimates and an acquisition function.

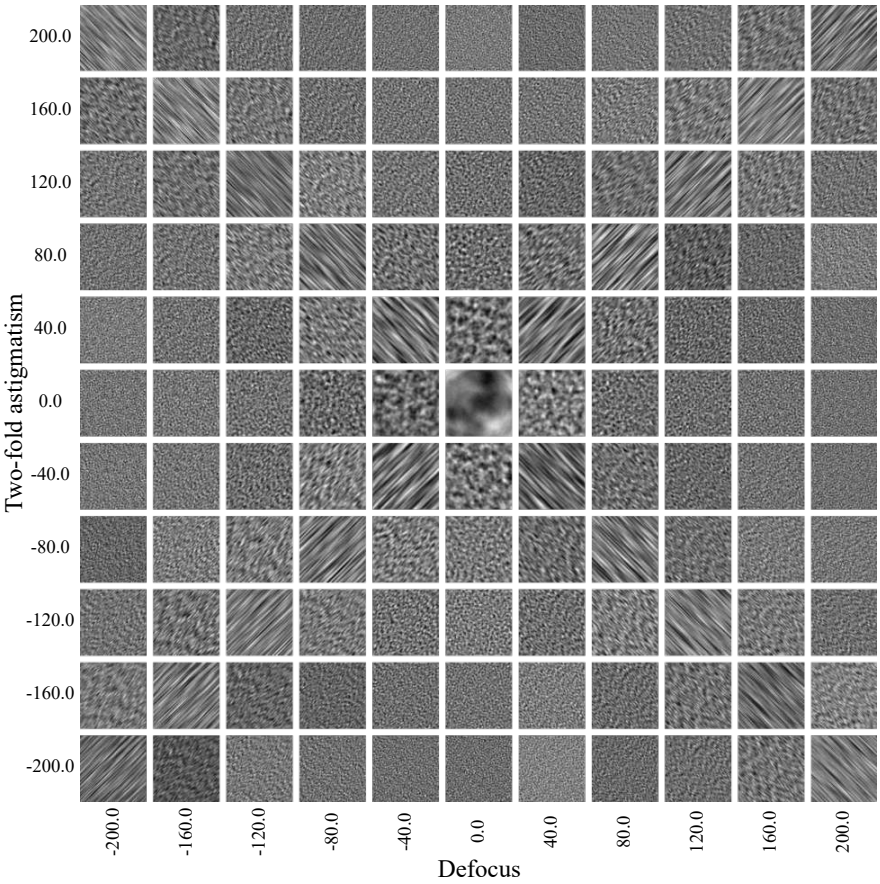


Figure 2.4. Ronchigram images produced by a high-fidelity simulation of the STEM for different defocus (C1) and two-fold astigmatism (A1x) values.

we take the logarithm of these values to highlight the edges of the Fourier shapes. This process yields

$$Y(t) := r(y(t)), \tag{2.3}$$

with $Y(t) \in \mathbb{R}^p$ representing the log power spectrum of the Ronchigrams and $r : \mathbb{R}^p \to \mathbb{R}^p$ representing the full preprocessing procedure. See Figures 2.4 and 2.5 for a comparison between $y(t)$ and $Y(t)$ of selected images. We use $Y(t)$ as the input for the neural network.

Even though the true aberration values are known for Ronchigrams generated by a simulator, it is impossible to infer the aberration values directly from the

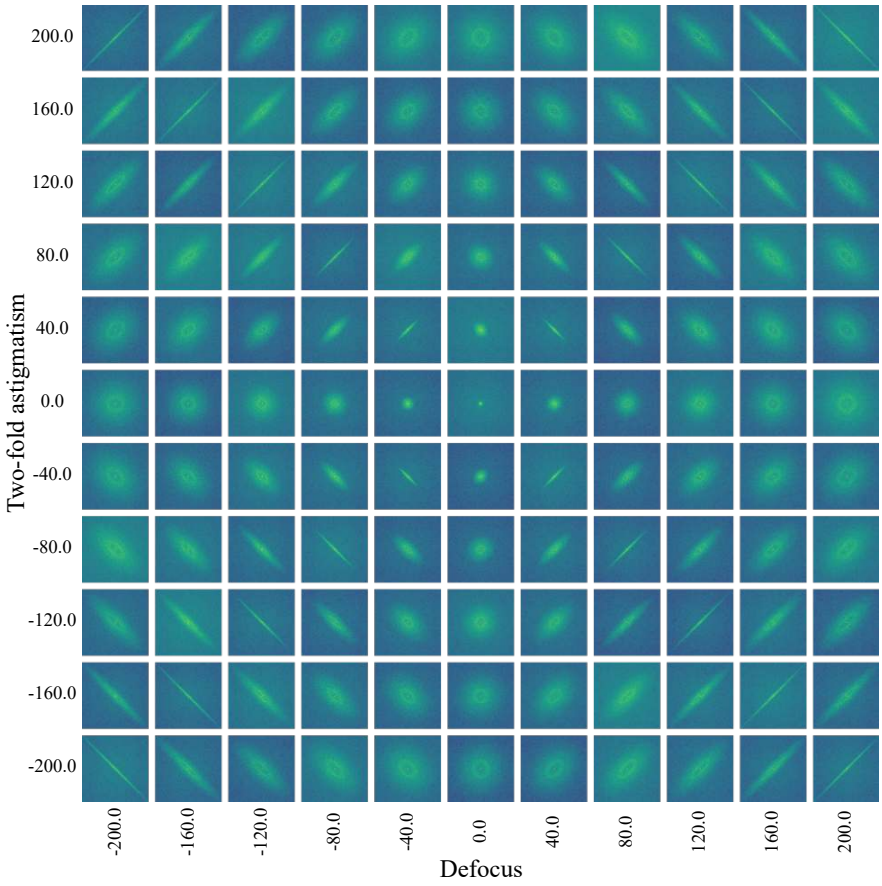


Figure 2.5. Preprocessed Ronchigram images corresponding to Figure 2.4, used to train the Ronchigram interpretation neural network.

Ronchigrams, since g in (2.1) cannot be inverted. However, it is possible to create a mapping from the preprocessed image to a (weighted) 2-norm of the aberrations. The reason is that the preprocessing makes the observation even in the aberrations: opposite aberration states x and $-x$ produce the same power spectrum $Y(t)$. Since $\|x\|_2 = \|-x\|_2$, this sign flip leaves the aberration norm unchanged, which makes the norm a consistent regression target. Examples of such power spectra can be seen in Figure 2.5. For this reason, we can train a neural network to identify aberration norms based on the large simulated dataset of pre-processed images

with known aberration values using the MSE loss function

$$L(\theta) = \frac{1}{N_S} \sum_{i=1}^{N_S} \left(\|x_i\|_2 - h(Y_i, \theta) \right)^2, \tag{2.4}$$

where θ represents the parameters of the neural network, $N_S := |\mathcal{D}_S|$ is the number of simulated Ronchigrams, and x_i and Y_i are the aberration state and preprocessed image of the i-th Ronchigram. The norms are normalized to approximately $[0, 1]$ across the dataset. After training on the simulated dataset, we obtain the optimized parameter vector θ^*. As mentioned before, we parametrize h based on the ResNet18 architecture, with a final layer added at the end to construct a scalar output, so $h : \mathbb{R}^p \to \mathbb{R}$. By substituting (2.1) into the learned neural network h, we can express the aberration 2-norm predictions as a function of the tuning knob settings through

$$\begin{aligned} z(t) &= h(Y(t), \theta^*), \\ &= h(r(y(t)), \theta^*), \\ &= h(r(g(\xi + u(t)) + \eta(t)), \theta^*), \\ &= f(u(t)) + \nu(t), \end{aligned} \tag{2.5}$$

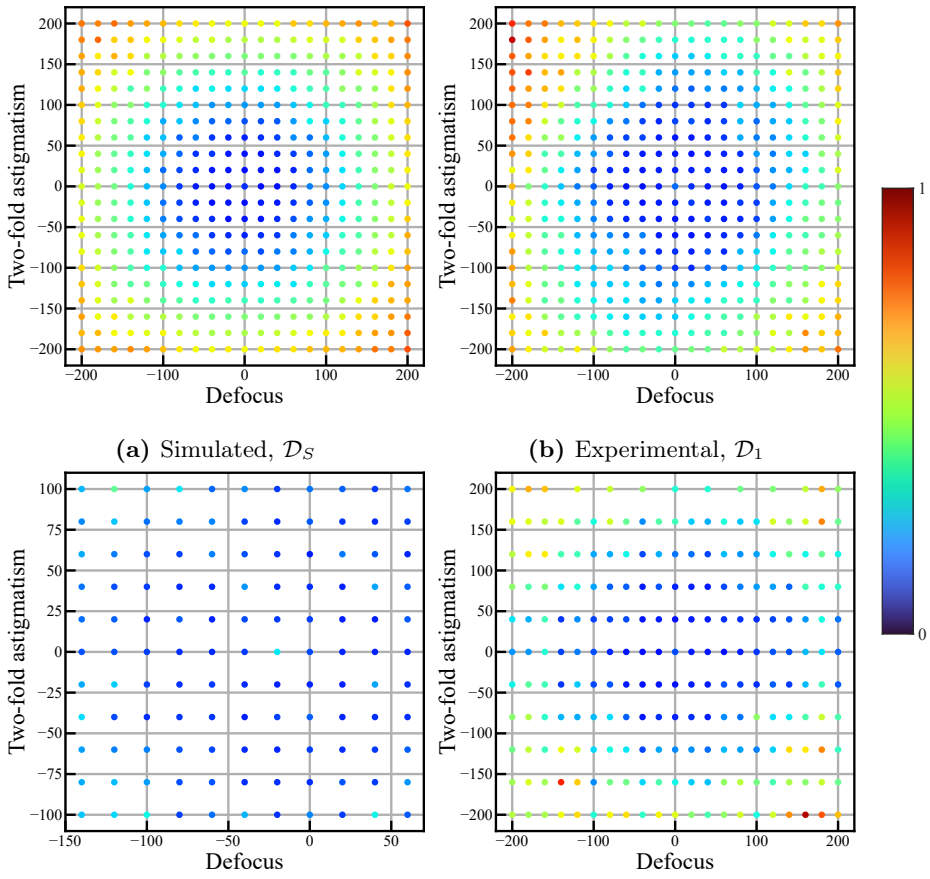
with $\nu \in \mathbb{R}$ assumed i.i.d. Gaussian noise with variance σ_ν^2. Note that the constants ξ and θ^* are omitted in the final expression. We perform transfer learning on the ResNet18 architecture, which is commonly done in image classification to leverage the network's pre-trained knowledge [77]. Figure 2.6 shows the norm predictions for different datasets using the trained network. The interpretation of the Ronchigrams yields a scalar value z that is close to zero when the EM is close to calibration, representing a much simpler relationship than the function g.

In the next section, we detail how to estimate the function f based on a limited number of Ronchigram observations using Gaussian process regression. Good estimates of f allow us to select the calibrating input more quickly.

2.3.2 Gaussian Processes

Gaussian processes (GPs) are a useful and versatile tool to estimate functions by incorporating prior beliefs and observed data [78]. The GP is characterized as a distribution of functions. Before considering any data, the prior distribution is completely characterized by a user-designed mean and covariance function. A key component of the GP design is this prior covariance function or kernel, which defines how function outputs correlate depending on the inputs:

$$k(u, u') = \operatorname{cov}(f(u), f(u')), \tag{2.6}$$



(a) Simulated, $\mathcal{D}_S$ **(b)** Experimental, $\mathcal{D}_1$

(c) Experimental, $\mathcal{D}_2$ **(d)** Experimental, $\mathcal{D}_3$

Figure 2.6. Normalized 2-norm predictions created by the trained neural network for a dataset of simulated Ronchigrams $\mathcal{D}_S$, and three datasets of experimental Ronchigrams labeled $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$. In every panel, the horizontal direction represents the defocus (C1) value, while the vertical direction represents the two-fold astigmatism (A1x) value.

where $k : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$, and in which $u, u' \in \mathcal{U}$ are the EM inputs. Note that the time index t is often omitted in this section to alleviate the notation. In addition to the kernel, the GP prior is characterized by a prior mean function $\bar{\mu} : \mathcal{U} \to \mathbb{R}$. Given a set of inputs $U \in \mathcal{U}^N$ and a set of corresponding cost values $F_U \in \mathbb{R}^N$, we

obtain the joint distribution

$$\begin{bmatrix} F_U \\ f(u) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \bar{\mu}_U \\ \bar{\mu}(u) \end{bmatrix}, \begin{bmatrix} \alpha & K_{Uu} \\ K_{uU} & k(u,u) \end{bmatrix} \right), \tag{2.7}$$

where $\alpha := K_{UU} + \sigma_\nu^2 I_N$, with σ_ν^2 the variance of the observation noise ν. The notation $F_X \in \mathbb{R}^N$ collects the noisy observations of the function at each element in the vector of inputs $X \in \mathcal{U}^N$, such that $F_X := [f(X_1) + \nu_1, f(X_2) + \nu_2, \ldots, f(X_N) + \nu_N]^\top$, and $\bar{\mu}_X \in \mathbb{R}^N$ collects the prior mean at those inputs, $\bar{\mu}_X := [\bar{\mu}(X_1), \bar{\mu}(X_2), \ldots, \bar{\mu}(X_N)]^\top$. The notation $K_{XY} \in \mathbb{R}^{N \times M}$ denotes the kernel matrix, where the element of the matrix in the i-th row and j-th column is calculated by $k(X_i, Y_j), i \in \{1, 2, \ldots, N\}, j \in \{1, 2, \ldots, M\}$, with $X \in \mathcal{U}^N, Y \in \mathcal{U}^M$. The joint distribution in (2.7) is conditioned on the dataset $\mathcal{D}(t)$ using Bayes' rule. The resulting posterior distribution is Gaussian and can be expressed in terms of a mean and variance for any input $u \in \mathcal{U}$ through

$$\begin{aligned} \mu\left(u \mid \mathcal{D}(t)\right) &:= \mathbb{E}\left[f(u) \mid \mathcal{D}(t)\right] \\ &= \bar{\mu}(u) + K_{uU}\alpha^{-1}(F_U - \bar{\mu}_U), \\ c\left(u, u' \mid \mathcal{D}(t)\right) &:= \operatorname{cov}\left(f(u), f(u') \mid \mathcal{D}(t)\right) \\ &= k(u, u') - K_{uU}\alpha^{-1}K_{Uu'}, \end{aligned} \tag{2.8}$$

where $\mu : \mathcal{U} \to \mathbb{R}$ and $c : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ are the mean and covariance estimates of the GP, respectively, and $\mathbb{E}[.]$ denotes the expected value. Here, we have an expression for the expected aberration 2-norm of any candidate input as well as a measure of the uncertainty. With an appropriate choice of prior covariance function, these estimates give a good indication of the calibration of the EM in just a few samples. Importantly, the calculation of the posterior distribution relies on the inversion of α, which is an N by N matrix. For this reason, the GP estimation scales poorly with large datasets ($O(N^3)$).

A crucial aspect of using Gaussian processes for regression is the design of the prior covariance function k. In this research, we opt for the widely used and simple squared exponential kernel, described by

$$k(u, u') = \sigma^2 \exp\left(-\frac{(u - u')^\top \Lambda^{-1} (u - u')}{2} \right), \tag{2.9}$$

where $\Lambda \in \mathbb{R}^{m \times m}$ and $\sigma \in \mathbb{R}_{\geq 0}$ are hyperparameters that represent the positive definite symmetric length scales and the signal standard deviation of the functions in the distribution. We use marginal likelihood hyperparameter optimization to tune these values based on the large simulated dataset of Ronchigrams, which is common practice in GPs [78, Chapter 5]. Afterward, the optimized hyperparameters are fixed during the online estimation of the experimental datasets. Here, we use the idea that the degree of smoothness is consistent between simulation and practice,

as well as between different days on the EM since we are always estimating the aberration 2-norm. Given a model that relates inputs u to predicted aberration 2-norms, we can move to the decision-making based on the model.

2.3.3 Bayesian Optimization

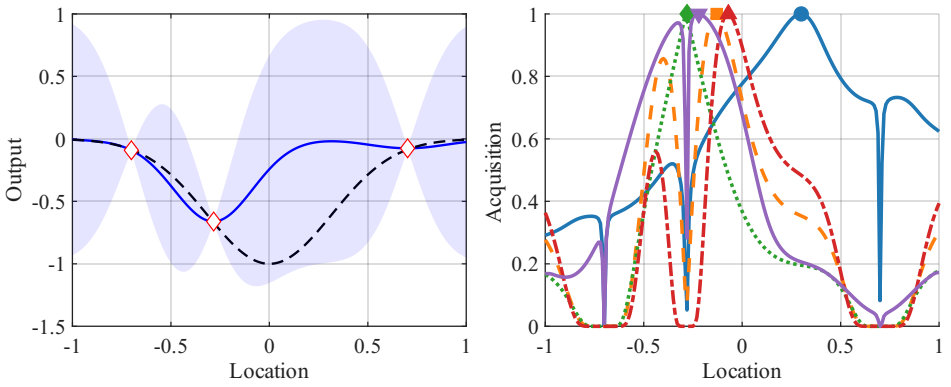
Bayesian optimization (BO) is an optimization method that can efficiently optimize black-box functions [72]. It makes use of two components. The first is the model, which usually comes in the form of a Gaussian process, as is the case in this research. The second component of BO is the acquisition function, which dictates how to select the next input based on the model. An example of an acquisition function is the probability of improvement (PI) [79], which can be evaluated for any candidate input $u \in \mathcal{U}$ through

$$\begin{aligned} \mathrm{PI}(u \mid \mathcal{D}(t)) :&= \mathrm{Prob}\left[f(u) < z^*\right] \\ &= \Phi\left(\frac{z^* - \mu(u \mid \mathcal{D}(t))}{\sqrt{c(u,u \mid \mathcal{D}(t))}}\right), \end{aligned} \tag{2.10}$$

where $z^* := \min_{u \in \mathcal{U}} \mu(u \mid \mathcal{D}(t))$ represents the current estimate of the lowest mean cost, and $\Phi : \mathbb{R} \to \mathbb{R}$ is the cumulative density function of a Gaussian variable. Standard acquisition functions in BO, such as PI, expected improvement (EI), and upper confidence bound (UCB) [80] typically aim to acquire low costs throughout the optimization process by balancing the exploration of uncertain regions of the model and the exploitation of low-cost regions of the model [72].

However, in this application only the calibration error of the finally recommended setting matters. The cost of the settings visited during the search is irrelevant, so this is a pure-exploration problem [81]. Acquisition functions that select queries to reduce the uncertainty about the optimum are designed for this regime. Entropy search reduces the uncertainty about the location of the optimum [82], max-value entropy search (MES) targets the optimal value [83], and the knowledge gradient (KG) selects the query that most improves the final recommendation [84], following the stepwise uncertainty reduction principle [85]. We follow the same principle, but measure the uncertainty about the calibrated state by the expected calibration error, expressed in the units of the calibration task. This quantity drives the exploration and, as detailed below, also supplies the stopping criterion. We estimate it by averaging, over a discrete set of candidate settings $\mathcal{U}_d \in \mathcal{U}^S$, the probability that each setting improves on the current best, weighted by its distance to the current best,

$$e(\mathcal{D}(t)) := \frac{1}{|\mathcal{U}_d|} \sum_{u \in \mathcal{U}_d} \mathrm{PI}(u \mid \mathcal{D}(t)) \, \|u - u^*\|_2, \tag{2.11}$$



(a) Gaussian process model **(b)** Acquisition functions

Figure 2.7. Gaussian process model in **(a)** (—) based on observations (◇) with 95% confidence interval and true function (- - -). The acquisition functions in **(b)** are based on that GP model, for the proposed method (—), EI (- - -), PI (····), MES (-·-·-), and KG (—). Each curve's maximizer is marked: the proposed method (●), EI (■), PI (◆), MES (▲), and KG (▼).

where $u^* := \arg\min_{u \in \mathcal{U}} \mu(u \mid \mathcal{D}(t))$ is the current best setting. Each candidate contributes its distance to u^*, weighted by the probability that it improves on u^*. The quantity $e(\mathcal{D}(t))$ therefore measures how far the recommended setting may still be from the true calibrating setting $-\xi$, given the data.

We select the next setting by its predicted influence on this quantity. Adding a candidate $\tilde{u} \in \mathcal{U}$ to the dataset changes the posterior covariance, and hence the PI of every point in $\mathcal{U}_d$, without requiring the corresponding observation, see (2.8). Writing $\mathrm{PI}(u \mid \mathcal{D}(t), \tilde{u})$ for the probability of improvement evaluated with this predicted covariance, the proposed acquisition function is

$$a(\tilde{u}) := -\frac{1}{|\mathcal{U}_d|} \sum_{u \in \mathcal{U}_d} \mathrm{PI}(u \mid \mathcal{D}(t), \tilde{u}) \, \|u - u^*\|_2. \tag{2.12}$$

The next setting is the candidate $\tilde{u} \in \mathcal{U}_d$ that maximizes a, equivalently the one whose observation is predicted to most reduce the expected calibration error.

The reason this acquisition function is particularly suitable for the problem setting is that it aims to disprove hypotheses that would have the worst impact on the calibration performance if they were correct. Here, we rely on the norm estimator typically producing approximately convex cost mappings f. Figure 2.7 illustrates the resulting behavior on an example GP model. The four existing acquisition functions, EI, PI, MES, and KG, all place their maximizer close to the current minimizer of the GP model to obtain low outputs; MES selects the

furthest of the four, but remains grouped with the others. The proposed acquisition function instead selects a setting far from the current minimizer, where a low observation would most change the recommended calibration. It thereby prioritizes the exploration of the hypotheses that would have the worst consequences.

The next section details a computationally efficient implementation of the covariance prediction.

Fast Computation of Predicted Variances

A prediction of the posterior covariance function for every candidate sampling point is required for the evaluation of the acquisition function described in the previous section. A naive implementation would add the current candidate point to the dataset of inputs, then evaluate c in (2.8). However, this would require an inversion of an $(N+1) \times (N+1)$ matrix for every candidate point. The computational cost of this implementation is $O(S(N+1)^3)$, with $S = |\mathcal{U}_d|$ the number of candidate points considered. Instead, we can use the following calculation for the change in the covariance depending on the next input setting $\tilde{u} \in \mathcal{U}$

$$\begin{aligned} \Delta(u, u') &:= c(u, u' \mid \mathcal{D}(t)) - c(u, u' \mid \mathcal{D}(t), \tilde{u}) \\ &= \begin{bmatrix} K_{Uu} \\ K_{\tilde{u}u} \end{bmatrix}^\top \begin{bmatrix} \mathcal{K} K_{\tilde{u}U} \alpha^{-1} & -\mathcal{K} \\ -\mathcal{K}^\top & \kappa^{-1} \end{bmatrix} \begin{bmatrix} K_{Uu'} \\ K_{\tilde{u}u'} \end{bmatrix} \end{aligned} \tag{2.13}$$

with $\mathcal{K} := \alpha^{-1} K_{U\tilde{u}} \kappa^{-1} \in \mathbb{R}^{N \times 1}$, and $\kappa := k(\tilde{u}, \tilde{u}) + \sigma_\nu^2 - K_{\tilde{u}U} \alpha^{-1} K_{U\tilde{u}} \in \mathbb{R}$. We obtain this expression by substituting the solutions for the covariances and applying the block matrix inversion formula.

The main computational efficiency gain comes from the fact that the expression above requires only a single matrix inversion (of the matrix α) for all candidate points, since α does not depend on the choice of the candidate point. Given α^{-1}, we only need to compute kernel slices and the inverse of the scalar κ to evaluate (2.13). For this reason, many different candidate points can be evaluated quickly once α^{-1} has been calculated. The computational cost of this implementation is $O(N^3 + S(N+1)^2)$, as we have to compute the $N \times N$ inverse and then S repetitions of $(N+1)$ matrix-vector products. The computational gain compared to $O(S(N+1)^3)$ is evident.

In the next section, we detail how the chosen acquisition function directly informs a stopping condition for the optimization algorithm.

Stopping Condition

The quantity (2.11) presents a problem-relevant value, which can be used to inform the BO algorithm when to stop optimizing. The quantity presents the expected value of the calibration error, which directly ties back to the problem formulated

in Section 2.2. For this reason, we can choose to stop the optimization once the expected calibration error is sufficiently low.

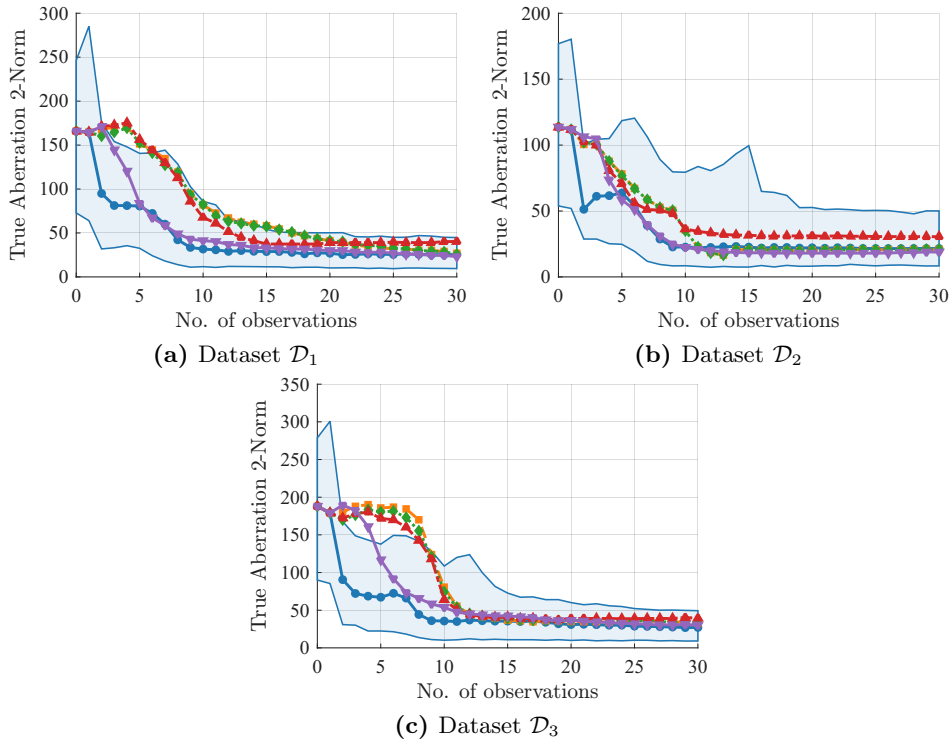
In the next section, we show the merit of the proposed method, which consists of the deep learning norm estimator, the Gaussian process, Bayesian optimization with a custom acquisition function, and the problem-specific stopping condition by applying it to experimental datasets gathered from an actual STEM. We compare it against four baseline acquisition functions, being the exploitation-driven EI and PI and the exploration-oriented MES and KG, whose definitions are given in Appendix 2.7.

2.4 Results

We attempt to calibrate the aberrations of defocus and two-fold astigmatism in two directions using the proposed method, hence $n = 3$. The prior mean function is taken constant, $\bar{\mu}(u) = 0.5$, since the interpreted cost is normalized to approximately $[0, 1]$ and no more specific prior information is available. To replicate an online implementation on the actual STEM machine, we use the following experimental setup. We start at an arbitrary initial Ronchigram image, which is taken randomly from one of the experimental datasets. We assume that the calibrated state is within a pre-specified distance for all aberrations. This determines the set $\mathcal{U}$. Next, we interpret the current Ronchigram image using pre-processing and deep learning. We then estimate the aberration norms for all points in $\mathcal{U}$ that have a corresponding image in the experimental dataset. We then evaluate the acquisition function (2.12) for these points and select the candidate that maximizes it, equivalently the one that minimizes the predicted expected calibration error, as our input. We then obtain a new Ronchigram from the experimental dataset and repeat. Since we have access to estimates of the aberrations for the experimental datasets, we can validate the calibration performance of our method on the experimental datasets.

We perform 500 Monte Carlo runs of the proposed method and of the four baseline acquisition functions, being EI, PI, MES, and KG, to show the merits of the problem-specific acquisition function. All methods share the same norm estimator and Gaussian process model, so the comparison isolates the effect of the acquisition function. To introduce randomness in each particular run, we consider a subset of the experimental datasets for each experiment. Specifically, we limit $\mathcal{U}$ while always ensuring that the calibrating input is within the input range. The randomization ensures that the calibrating input is not always in the exact center of the dataset, creating varied curves that are more representative of the online implementation.

The results are shown in Figures 2.8 and 2.9. The main benefit of the proposed method is early convergence. It reaches a median true norm below 100 nm by the third observation, where the baselines need five to eleven, and the norm it attains



(a) Dataset $\mathcal{D}_1$ **(b)** Dataset $\mathcal{D}_2$

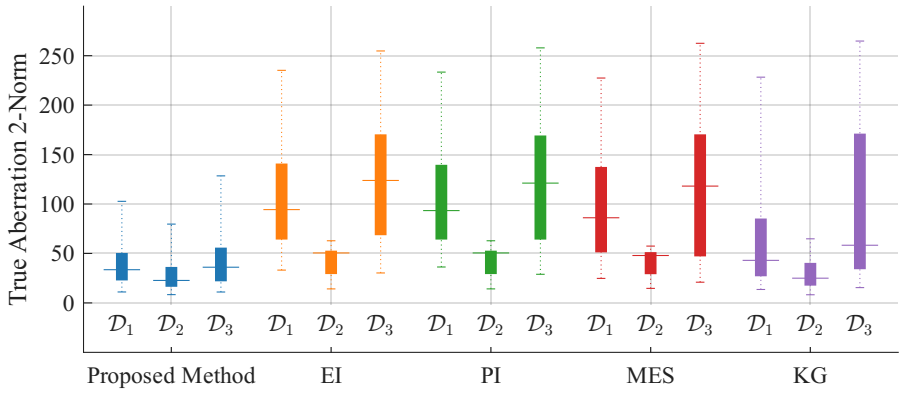
(c) Dataset $\mathcal{D}_3$

Figure 2.8. True aberration norm of the GP-minimizing input at each iteration for the proposed method (—●—), EI (- -■- -), PI (·-◆-·), MES (- -▲- -), and KG (—▼—). Plotted are the medians of 500 Monte Carlo runs, together with the 90th percentile band of the proposed method. The proposed method and KG reach the lowest norms, and the proposed method does so in the fewest observations.

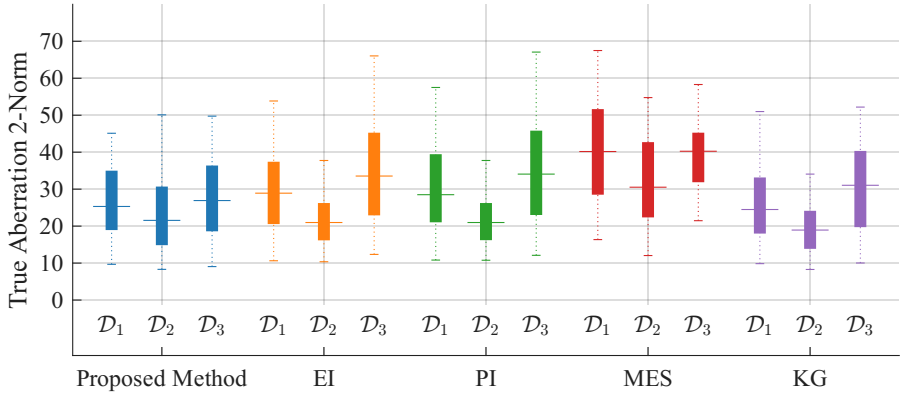
at observation 10 takes EI and PI another 12 to 26 observations to match. By observation 30 the methods largely converge, with the proposed method and KG reaching the lowest norms. KG performs well for the same reason as the proposed method: both select settings that improve the recommended optimum rather than settings with low predicted cost, which is what the calibration objective rewards. None of the methods reach zero, because the norm estimator was trained on a simulator that does not perfectly match the experimental data. This mismatch also makes the stopping quantity optimistic: the expected calibration error falls below 1 nm within about eleven observations, whereas the true median norm is then still around 30 nm. The quantity e measures the uncertainty of the Gaussian process, so it cannot detect a bias shared by all of its predictions. It therefore signals when further observations stop helping, rather than certifying a specific accuracy.

Table 2.1. Comparison of existing automated STEM aberration calibration methods and the method proposed in this chapter. Highlighted are the aberration types compensated, their possible initial values, the calibration accuracy typically achieved with the method, the runtime, and the requirements on the specimen. [1]These methods have operating modes that can calibrate higher-order aberrations as well, but only the C1/A1 mode is considered here for a fair comparison. [2]We may assume that 200 nm for both defocus (C1) and two-fold astigmatism (A1) is possible. [3]These papers only report precision, not accuracy. [4]Based on full STEM images, so does not use any Ronchigrams. Such images take longer to acquire. [5]Includes time needed for lens deGaussing.

Method	**Aberr.**	**Initial conditions**	**Calibration accuracy**	**Time taken**	**Specimen type**
CEOS (DCOR) [47]	C1, A1[1]	Not reported[2]	$\lvert C1\rvert < 10\,nm$ $\lvert A1\rvert < 10\,nm$	1 minute[4]	stable, high-contrast, radiation-resistant amorphous or crystalline
OptiSTEM(+) [46]	C1, A1[1]	Not reported[2]	$\lvert C1\rvert < 10\,nm$ $\lvert A1\rvert < 10\,nm$	1 minute[4]	
BEACON [51]	C1, A1[1]	$\lvert C1\rvert < 5\,nm$ $\lvert A1\rvert < 5\,nm$	Not reported[3]	15 s (HfO_2) or 45 s (Au NP) (20 or 57 Ronchigrams)	HfO_2 and Au NP
Ma et al. [50]	C1, A1, B2, A2, C3	$\lvert C1\rvert < 25\,nm$ $\lvert A1\rvert < 40\,nm$	Not reported[3]	240 s (50 Ronchigrams)[5]	amorphous
This chapter	**C1, A1**	**$\lvert C1\rvert < 200\,nm$** **$\lvert A1\rvert < 200\,nm$**	**$\lvert C1\rvert < 15\,nm$** **$\lvert A1\rvert < 20\,nm$**	**15 s (20 Ronchigrams)**	**amorphous**



(a) Observation 10



(b) Observation 30

Figure 2.9. Box plot of the aberration norm for the proposed method (■), EI (■), PI (■), MES (■), and KG (■) for 500 Monte Carlo runs. Within each method, the three boxes are $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$. The median aberration norm for each set of experiments is marked with a horizontal line. The 50th and 90th percentiles are represented by the solid bar and dotted lines, respectively. At observation 10 the proposed method already reaches the lowest norms, and by observation 30 the proposed method and KG have the lowest medians while MES has the highest.

Table 2.1 shows a performance comparison to existing methods found in the literature. Specifically, we consider DCOR by CEOS [47] and OptiSTEM(+) by Thermo Fisher Scientific [46], which are the most widely used automated calibration methods. Both rely on full STEM images, rather than Ronchigrams, increasing the time needed for calibration. Second, we consider BEACON [51], which uses BO of the normalized image variance, a value which correlates with

the presence of aberrations. Third, we consider the method by Ma et al. [50], which uses BO of the beam emittance to improve all aberrations simultaneously. While the existing methods can all calibrate higher-order aberrations as well, we consider the mode in which they calibrate only lower-order aberrations in the comparison. We reiterate that the lower-order aberrations require the most frequent recalibration, which is why they are the focus of this chapter. Note that some of the performance metrics are omitted or estimated from the cited literature, since precise data on these statistics is unavailable. From the table, it is apparent that the proposed method is among the fastest to calibrate the STEM. The main reason for the increase in speed is the shift from using full STEM images to using Ronchigrams. Furthermore, our method has been validated on real data by using OptiSTEM(+) to determine the real aberration values and by comparing against DCOR's corrections. In this setting, we achieve similar calibration performance to these industry standards. Additionally, since our method is purely data-driven, the performance can be enhanced in a straightforward manner by improving the simulator or by expanding the simulated dataset $\mathcal{D}_S$, unlike the physics-based methods DCOR and OptiSTEM(+).

2.5 Conclusion

In this chapter, we proposed a fully data-driven method to calibrate aberrations in the STEM. The proposed method separates the calibration process into two steps, being the interpretation and decision-making steps. The interpretation is performed through an adapted ResNet18 deep learning architecture, which assigns a scalar performance value to the current microscope image or Ronchigram. The neural net is trained on a large database of simulated Ronchigrams, due to the limited availability of experimental images. The decision-making utilizes a Gaussian process model of the mapping from calibration settings to the scalar performance value, which drives online Bayesian optimization with a tailored acquisition function. In the spirit of entropy search, this acquisition function explores to reduce the uncertainty about the calibrated state, and it supplies a stopping criterion tied to the expected calibration error. We showed the merits of the proposed pipeline and acquisition function on real STEM data. High calibration accuracy and consistency are obtained, more quickly than existing methods due to the fact that Ronchigrams can be obtained much faster than full STEM images. The present research shows the potential of machine learning in STEM calibration in order to alleviate reliance on experts for STEM aberration correction.

2.6 Discussion

The main limitation of the current method is the reliance on simulated data for two reasons. Firstly, the simulator inherently has some model mismatch with the real EM setup, resulting in a neural network and consequently in a Gaussian process model which are not entirely accurate. Second, the trained neural network is not capable of extrapolating to data it has not seen before, such as other materials or other aberration types. Both of these shortcomings can be reduced by extending the simulator. Note that the simulators are developed to approximate the Ronchigrams obtained from the EM as well as possible. They are validated using real STEM Ronchigrams before being considered suitable to capture the STEM image generation. Therefore, the accuracy of current simulators in the context of STEM can often be assumed, and with it the applicability of the methods proposed here. The future development of more and more accurate simulators can further mitigate this gap between simulations and actual STEMs.

Future work includes extending the number of aberration types to be compensated. The proposed method can potentially generalize to more aberration types, as long as the simulator can produce representative Ronchigrams. Typically in these settings, the images are preprocessed using so-called patching before taking the Fourier transform to retain local frequency information [86, Chapter 3]. We hypothesize that the increase in dimensionality coincides with a decrease in calibration accuracy or speed. The main hurdle in this aspect is the model mismatch between the simulator and the real STEM, which causes the norm estimates to be somewhat less informative.

The two main limitations identified above, the scalar representation and the simulator-to-reality model mismatch, are addressed directly in the next chapter. Chapter 3 replaces the scalar norm estimate with a multi-dimensional representation of the Ronchigram learned by a variational autoencoder, so that less information is discarded as the number of aberrations grows. It also estimates the model mismatch from a small amount of real data rather than ignoring it.

2.7 Appendix: Baseline Acquisition Functions

Section 2.4 compares the proposed acquisition function against four baselines. All of them use the same Gaussian process posterior (2.8), hyperparameters, and discrete input set $\mathcal{U}_d$ as the proposed method, so the comparison isolates the effect of the acquisition function. Let $s(u) := \sqrt{c(u, u \mid \mathcal{D}(t))}$ denote the posterior standard deviation and $f^* := \min_{u \in \mathcal{U}} f(u)$ the unknown optimal cost.

Expected improvement and probability of improvement are the exploitation-driven baselines. Both are evaluated with an exploration margin $\delta = 0.5$ that lowers the incumbent to $z^* - \delta$ and thereby requires a minimum improvement. Probability

of improvement is defined in (2.10) [79]. Expected improvement additionally weights the improvement by its magnitude [87],

$$\mathrm{EI}(u) = s(u)\,(\beta(u)\,\Phi(\beta(u)) + \phi(\beta(u)))\,, \tag{2.14}$$

with $\beta(u) := (z^* - \delta - \mu(u \mid \mathcal{D}(t)))/s(u)$, and ϕ and Φ the standard normal density and distribution functions. The setting that maximizes EI is selected.

Max-value entropy search selects the setting whose observation is most informative about f^* [83]. Adapted to the minimization considered here, its acquisition function is

$$\mathrm{MES}(u) = \frac{1}{K}\sum_{k=1}^{K}\left[\frac{\gamma_k(u)\,\phi(\gamma_k(u))}{2\,\Phi(\gamma_k(u))} - \log\Phi(\gamma_k(u))\right], \tag{2.15}$$

with $\gamma_k(u) := (\mu(u \mid \mathcal{D}(t)) - f_k^*)/s(u)$, in which $\{f_k^*\}_{k=1}^K$ are samples of the optimal cost drawn from the posterior over $\min_{u\in\mathcal{U}_d} f(u)$. The setting that maximizes $\mathrm{MES}(u)$ is selected.

The knowledge gradient selects the setting whose observation is expected to most reduce the best posterior mean [84],

$$\mathrm{KG}(u) = \min_{u'\in\mathcal{U}_d}\mu(u' \mid \mathcal{D}(t)) - \mathbb{E}\left[\min_{u'\in\mathcal{U}_d}\mu(u' \mid \mathcal{D}(t)\cup\{(u,z)\}) \,\middle|\, \mathcal{D}(t)\right], \tag{2.16}$$

where z is the predicted observation at u and the expectation is taken over its Gaussian posterior. The posterior mean after this hypothetical observation is affine in z, so the expectation admits a closed-form evaluation [84]. The posterior covariance required for the update is obtained from the same prediction (2.13) used by the proposed method.

3

Bridging the Simulation-to-Reality Gap in Electron Microscope Calibration via VAE-EM Estimation

Abstract - Electron microscopy has enabled many scientific breakthroughs across multiple fields. A key challenge is the tuning of microscope parameters based on images to overcome optical aberrations that deteriorate image quality. This calibration problem is challenging due to the high-dimensional and noisy nature of the diagnostic images, and the fact that optimal parameters cannot be identified from a single image. We tackle the calibration problem for Scanning Transmission Electron Microscopes (STEM) by employing variational autoencoders (VAEs), trained on simulated data, to learn low-dimensional representations of images, whereas most existing methods extract only scalar values. We then simultaneously estimate the model that maps calibration parameters to encoded representations and the optimal calibration parameters using an expectation maximization (EM) approach. This joint estimation explicitly addresses the simulation-to-reality gap inherent in data-driven methods that train on simulated data from a digital

This chapter is based on van Hulst et al. [88].

twin. We leverage the known symmetry property of the optical system to establish global identifiability of the joint estimation problem, ensuring that only one combination of state and model explains the observations. We demonstrate that our approach is substantially faster and more consistent than existing methods on a real STEM, achieving a 2× reduction in estimation error while requiring fewer observations. This represents a notable advance in automated STEM calibration and demonstrates the potential of VAEs for information compression in images. Beyond microscopy, the VAE-EM framework applies to inverse problems where simulated training data introduces a reality gap and where non-injective mappings would otherwise prevent unique solutions.

3.1 Introduction

Electron microscopy enables atomic-resolution imaging with applications across materials science, life sciences, and the semiconductor industry. Calibrating these microscopes requires frequent adjustments to compensate for optical aberrations that deteriorate image quality. Calibration is performed by analyzing special diagnostic images called Ronchigrams, which are diffraction patterns whose features reveal information about the microscope's current aberration state [68]. Currently, the calibration is typically performed or monitored by experts who interpret these patterns and determine appropriate corrections, making calibration expensive and time-consuming. For this reason, automating the calibration process is highly desirable.

The challenge in automating the calibration process comes from the complex, non-injective relationship between aberration parameters and the Ronchigram [62]. Different aberration states can produce indistinguishable diffraction patterns. This ambiguity, combined with the high dimensionality of Ronchigram images (typically around 10^5 pixels) and the high sensitivity of the microscope to environmental factors, makes it difficult to determine the optimal calibration parameters.

While significant progress has been achieved over the past decade in the automation of STEM calibration, it is still an area of active research. Existing approaches broadly fall into two categories: physics-based and data-driven methods. Physics-based approaches were developed first, employing physical optics models to relate Ronchigram features to aberration parameters [44–47, 70]. These methods are generally robust to different operating conditions, but are fundamentally limited by the accuracy of the underlying physical models and by the inability to adapt to day-to-day variations in real electron microscopes.

In contrast, data-driven calibration methods remove the reliance on detailed physics models (see [62] for a comprehensive review). These methods typically employ deep learning to produce a scalar quality metric for each image that can be optimized [48–51]. Optimizing such a scalar value is straightforward, but has its limitations. The scalar goal rapidly loses information content as the number of calibration parameters increases. Additionally, the referenced data-driven methods are trained on simulated data (due to limited experimental data availability), creating a simulation-to-reality gap when deployed on real microscopes. Despite these limitations, data-driven methods remain more promising for achieving robust calibration across different microscopes and operating conditions, as they can adapt to day-to-day variations without requiring accurate physical models.

Although the data-driven calibration methods developed thus far have shown promising results, the reliance on scalar representations and simulation data fundamentally limits their performance and applicability. This chapter aims to address these two limitations directly. First, we introduce multi-dimensional representations

through variational autoencoders (VAEs) [89]. Second, we adopt a simultaneous state and model estimation approach based on expectation maximization (EM) [90] to explicitly account for the simulation-to-reality gap. This combination of VAE-based dimensionality reduction with EM-based joint estimation is, to our knowledge, novel in the context of inverse problems. Additionally, we introduce the use of symmetry-constrained kernels to establish global identifiability, which presents a theoretical contribution that addresses a fundamental limitation of standard EM approaches for inverse problems. We model the map from calibration parameters to VAE latent representations using Gaussian processes (GPs) [78]. GPs provide a flexible, non-parametric framework that can embed prior knowledge of the problem structure. The key insight that enables the joint state and model estimation approach is that the aberrations exhibit a known symmetrical structure [3]. By enforcing these symmetries in the GP model, we obtain global identifiability of the joint estimation problem, a property that is typically absent in standard EM approaches for inverse problems [31, 91]. Additionally, we employ an adaptive refinement strategy for the discrete set of aberration states considered in the EM algorithm that improves estimation accuracy even when considering a small number of aberration state candidates. This refinement strategy takes inspiration from particle filtering methods [92]. Finally, we develop an input selection strategy that chooses calibration inputs to maximize information gain to accelerate convergence.

Our contributions can be summarized as follows:

1. We introduce variational autoencoders in electron microscopy to learn multi-dimensional latent representations of Ronchigram images. These representations capture more information than scalar quality metrics used in prior work.

2. We develop a novel VAE-EM framework for joint state and model estimation that addresses the simulation-to-reality gap inherent in training on simulated data. We establish global identifiability of this joint estimation problem by exploiting symmetry constraints from Fourier optics.

3. We validate on real STEM data, showing that the proposed method more than halves the estimation error of existing automated calibration methods and converges to accurate estimates within tens of seconds.

While developed for STEM calibration, the methodological contributions are of general interest and extend beyond this application domain. The VAE-EM framework and identifiability analysis apply to inverse problems where known structural constraints (such as evenness) can be exploited to restore unique solutions in joint state-model estimation. We discuss the conditions for general applicability in Section 3.6.

The remainder of this chapter is organized as follows. Section 3.2 provides the problem formulation, including the STEM imaging model and preprocessing steps. Section 3.3 details the proposed VAE architecture and training procedure. Section 3.4 presents the EM algorithm for joint state and model estimation, along with identifiability analysis. Section 3.5 showcases experimental results on real STEM data, and Section 3.6 discusses the implications of our findings.

3.2 Problem Formulation

The electron microscope calibration problem and experimental setup are presented in detail in our previous work [62]. In short, the problem involves overcoming optical aberrations in scanning transmission electron microscopy (STEM), as these aberrations deteriorate image quality. This is achieved by interpreting high-dimensional Ronchigram images to determine the calibration parameters that result in the best image quality. In this section, we summarize the key aspects and structure of the calibration problem including the aberration model, Ronchigram observations and preprocessing steps.

3.2.1 Aberration Optics and Ronchigrams

Let the microscope aberration state $x(t) \in \mathbb{R}^n$ represent the aberration levels at discrete time $t \in \mathbb{N}$, where $x(t) = 0$ corresponds to perfect calibration. The aberrations evolve according to:

$$x(t+1) = x(t) + u(t), \tag{3.1}$$

where $u(t) \in \mathbb{R}^n$ represents the user-selected calibration inputs applied to compensate aberrations. Over typical calibration timescales, aberrations remain constant unless nonzero inputs are applied. This implies that the evolution of the aberrations is deterministic for our purposes. Additionally, if the aberration state becomes known, we can immediately achieve optimal calibration by applying the input $u(t) = -x(t)$. To alleviate notation, we denote the cumulative input up to time t as $s(t) = \sum_{\tau=0}^{t-1} u(\tau)$, so that $x(t) = x_0 + s(t)$.

The STEM produces high-dimensional Ronchigram observations $y(t) \in \mathbb{N}_0^p$ (where $\mathbb{N}_0 = \{0, 1, 2, \ldots\}$) representing electron counts at p pixels (with p typically in the order of tens of thousands). For a fixed sampling location and beam intensity, the observations follow a Poisson distribution. The rate of this distribution is then determined by the aberration state $x(t)$ through an unknown image formation function $g : \mathbb{R}^n \to \mathbb{R}_+^p$ (where $\mathbb{R}_+ = (0, \infty)$). We model

$$y(t) \sim \text{Poisson}\left(g(x(t))\right), \tag{3.2}$$

where the Poisson distribution applies element-wise to each pixel independently.

We employ two datasets of Ronchigram images in this work. The first is a large simulated dataset generated using a high-fidelity wave optics simulator based on the multi-slice method [93] that contains many different aberration values. This dataset is used for VAE training, which requires substantial data volumes. While the simulator is relatively accurate, it inherently suffers from undermodeling and does not capture day-to-day variations present in the experimental microscope. The second dataset consists of experimental Ronchigrams collected from a real STEM system on separate operating days. In both datasets, the aberration states are known since they are set by the user. These states can be used to validate our calibration procedure by comparing estimated aberrations to ground truth values. Importantly, the known aberration values in the simulated dataset also enable us to optimize hyperparameters such as the prior mean and covariance functions that will be used for Gaussian process regression in Section 3.4.

3.2.2 Preprocessing and Symmetry of the Ronchigrams

To extract useful information from the raw Ronchigram images $y(t)$, we perform several preprocessing steps. These preprocessing steps follow standard practices in automated STEM calibration and are laid out in detail in our previous work [62]. To summarize, the workflow consists of three operations applied to each image vector $y(t)$:

1. Apply a Hann window element-wise to reduce spectral leakage.
2. Transform to the 2D Fourier domain by treating the p-dimensional vector as an image with spatial pixel coordinates.
3. Compute the power spectrum by taking the element-wise squared modulus.

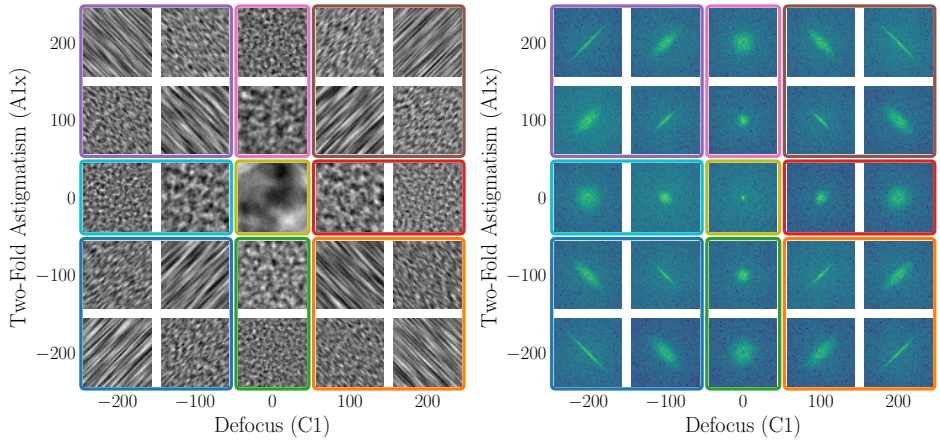
We denote the composite preprocessing operator by $\mathcal{P}$, so that the preprocessed observations are

$$\tilde{y}(t) = \mathcal{P}\{y(t)\}, \tag{3.3}$$

where $\mathcal{P} : \mathbb{N}_0^p \to \mathbb{R}_+^p$ includes windowing, the 2D discrete Fourier transform, and the squared modulus operation. The technical definition of the 2D Fourier transform in terms of spatial coordinates and frequency indices is provided in Appendix 3.7. After preprocessing, the observations $\tilde{y}$ can be modeled as a noisy function of the aberrations

$$\tilde{y}(t) = h(x(t)) + \eta(t), \tag{3.4}$$

where $h(x) := \mathbb{E}[\mathcal{P}\{y\} \mid x]$ denotes the expected power spectrum as a function of



(a) Ronchigram images **(b)** Fourier power spectra

Figure 3.1. Grid of Ronchigram images and their Fourier power spectra for different aberration values. In both panels, the horizontal axis varies defocus ($C1$) from -200 to $+200$ nm, while the vertical axis varies two-fold astigmatism ($A1x$) over the same range. The center images correspond to well-tuned aberrations ($C1 = 0$, $A1x = 0$). Colored rectangles group images in specific aberration regions. Note that the Fourier power spectra in **(b)** exhibit symmetry about the origin, with images at (x_1, x_2) and $(-x_1, -x_2)$ appearing similar (e.g., purple and orange regions look similar, cyan and red regions look similar).

the aberration state and $\eta(t) := \tilde{y}(t) - h(x(t))$ captures the measurement noise. The distribution of η is typically non-trivial due to the nonlinear preprocessing steps and the Poisson noise, but we do not require specific distributional assumptions for η at this point. Figure 3.1 shows example Ronchigram images and their preprocessed Fourier power spectra for different aberration values. A key result of the preprocessing is that h is even with respect to the aberrations:

$$h(-x) = h(x) \quad \forall x \in \mathbb{R}^n. \tag{3.5}$$

This symmetry arises from the physics of electron optics and appears only in the Fourier power spectrum, not in the spatial domain (thus, in general, $g(-x) \neq g(x)$). The full derivation of this property from first principles is provided in Appendix 3.7. This derivation uses continuous Fourier analysis in the spatial frequency domain to establish the symmetry for the continuous intensity function, then shows that the discrete 2D Fourier transform preserves this symmetry after sampling onto a pixel grid. The symmetry property is central to our approach, as it enables unique identifiability in the joint estimation problem developed in Section 3.4.

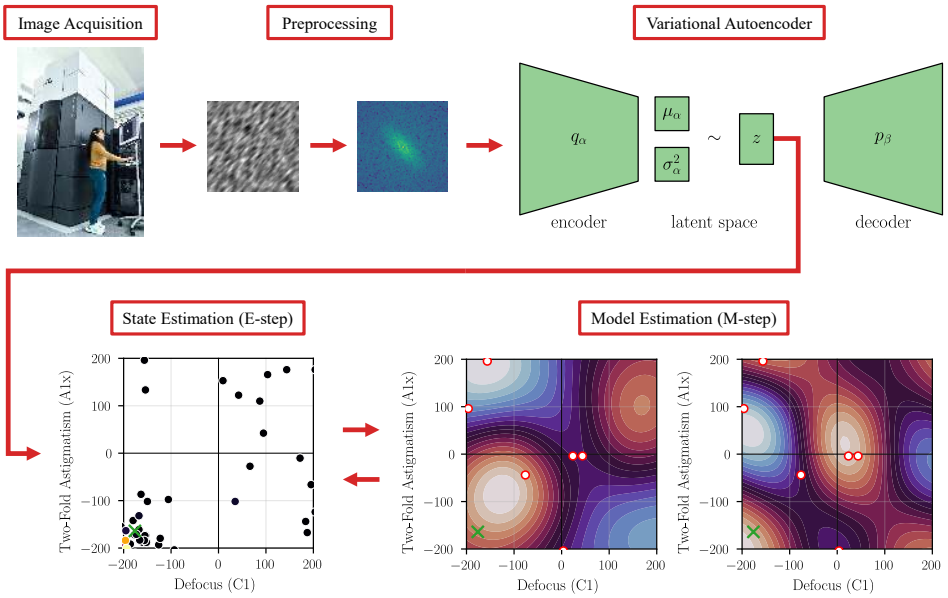


Figure 3.2. Calibration pipeline overview. High-dimensional Ronchigram images are preprocessed and then encoded to a low-dimensional latent space using a VAE. The EM algorithm then iteratively refines both the aberration state estimate and the mapping from aberrations to latent representations. In the E-step, the distribution over aberration states is estimated by evaluating the likelihood of observed latent space values for different candidate states. These candidates (●●●) are highlighted in the aberration space, where the color indicates their estimated likelihood (low ● to high ●). The × indicates the ground truth aberration state. Note that this state and its symmetrical neighbor have an increased density of candidates due to the adaptive candidate refinement strategy. The most likely candidates are also clustered around the ground truth state. The M-step then updates the Gaussian process model of the aberration to latent space mapping. The current model belief based on 5 data points (○) is illustrated as a function of the aberrations. Note that this model belief is symmetric due to the symmetry constraint in the model structure. The symmetry point is not necessarily at the origin (in the true aberration coordinates) due to the fact that the state estimate is not yet perfect. The M- and E-steps are iterated until convergence.

3.2.3 Approach

Our goal is to calibrate the STEM by making the aberration state x zero. This goal can be equivalently characterized as the estimation of the aberration state

$x(t)$ from the observations $\{y(t)\}_{t=0}^{T}$ for corresponding chosen inputs $\{u(t)\}_{t=0}^{T-1}$, since we have full knowledge of the dynamics and control inputs. Additionally, estimating the full trajectory $\{x(t)\}_{t=0}^{T}$ is equivalent to estimating the initial state $x_0 = x(0)$ that then determines all future states via the deterministic dynamics $x(t) = x_0 + s(t)$.

The estimation problem is challenging for two fundamental reasons. First, the Ronchigram observations $y(t) \in \mathbb{R}_+^p$ are extremely high-dimensional (typically $p \approx 10^5$ pixels), making direct estimation computationally intractable and statistically inefficient. Second, the image formation function $g : \mathbb{R}^n \to \mathbb{R}_+^p$ (and similarly $h : \mathbb{R}^n \to \mathbb{R}_+^p$) is unknown and varies day-to-day due to unmodeled effects, environmental conditions, and instrument drift. While physics-based models exist, they are insufficiently accurate for precise and adaptive calibration. Additionally, the choice of inputs $\{u(t)\}_{t=0}^{T-1}$ can significantly affect the estimation convergence rate, so input selection is an important consideration that we address.

To address these challenges, we propose a two-stage approach:

1. **Dimensionality reduction (Section 3.3):** Use a variational autoencoder (VAE) to learn a low-dimensional latent representation $z(t) \in \mathbb{R}^\ell$ (where $\ell \ll p$) of the preprocessed observations $\tilde{y}(t)$ that preserves as much information about $x(t)$ as possible. As will be shown in the next section, this results in a simplified observation model $z(t) = f(x(t)) + \nu(t)$ where $f : \mathbb{R}^n \to \mathbb{R}^\ell$ is the effective latent mapping, and $\nu(t)$ is the noise whose characteristics are discussed in Section 3.3.

2. **Joint state and model estimation (Section 3.4):** Given the latent observations and chosen inputs $\{z(t)\}_{t=0}^{T}, \{u(t)\}_{t=0}^{T-1}$, simultaneously estimate both the initial state x_0 and the latent mapping f using an Expectation-Maximization (EM) algorithm with provable identifiability guarantees.

The full calibration pipeline proposed in this chapter is illustrated in Figure 3.2. In our previous work, we also proposed a two-stage approach consisting of Ronchigram interpretation and decision-making based on the interpretations [62]. However, this approach relied on scalar image representations and had no explicit model estimation step.

3.3 Learning Compact Image Representations with VAEs

We employ a variational autoencoder (VAE) to learn a low-dimensional latent representation $z \in \mathbb{R}^\ell$ (where $\ell \ll p$) of the preprocessed Ronchigram power spectra $\tilde{y}$.

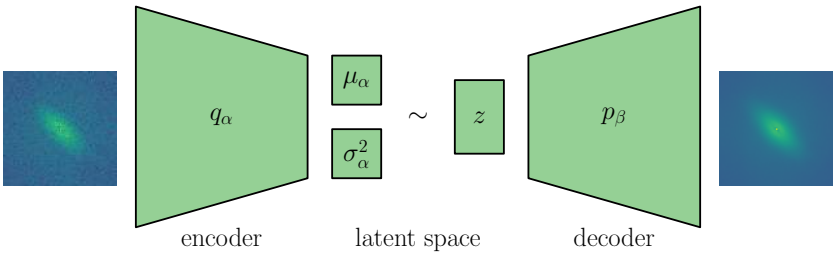


Figure 3.3. Variational Autoencoder architecture. The encoder maps preprocessed Ronchigram power spectra to a low-dimensional latent distribution, from which samples are decoded to reconstruct the input. The weights and biases of the encoder and decoder networks are optimized to maximize the ELBO objective in (3.7), which balances reconstruction accuracy and latent space regularization. The example reconstruction to the right of the decoder resembles the input and shows the denoising effect of the VAE.

3.3.1 VAE Architecture

The VAE consists of an encoder network $q_\alpha(z|\tilde{y})$ and decoder network $p_\beta(\tilde{y}|z)$

$$\begin{aligned} q_\alpha(z|\tilde{y}) &= \mathcal{N}(z; \mu_\alpha(\tilde{y}), \sigma^2_\alpha(\tilde{y}) I_\ell), \\ p_\beta(\tilde{y}|z) &= \mathcal{N}(\tilde{y}; \mu_\beta(z), \sigma^2_\beta I_p), \end{aligned} \tag{3.6}$$

where α and β are the encoder and decoder parameters, respectively. The encoder maps the input $\tilde{y}$ to a Gaussian distribution in the latent space with mean $\mu_\alpha(\tilde{y})$ and diagonal covariance $\sigma^2_\alpha(\tilde{y})I_\ell$. The decoder reconstructs the input from latent samples z using a Gaussian likelihood with mean $\mu_\beta(z)$ and fixed variance $\sigma^2_\beta I_p$. See Figure 3.3 for a visualization of the VAE architecture.

We implement both encoder and decoder using convolutional neural networks (CNNs) to process the Ronchigram images efficiently. The encoder consists of five convolutional layers with progressively increasing channel dimensions $[16, 32, 64, 128, 256]$, each followed by batch normalization and LeakyReLU activation (negative slope 0.01). Strided convolutions (stride 2) perform downsampling. The final convolutional output is flattened and passed through fully connected layers to produce the latent mean $\mu_\alpha(\tilde{y})$ and log-variance $\log \sigma^2_\alpha(\tilde{y})$. The decoder mirrors this architecture with transposed convolutions for upsampling. We set the latent dimension $\ell = n$ to match the number of aberration parameters. The symmetry constraint $h(-x) = h(x)$ means the encoder must map both x and $-x$ to the same latent representation, but this identification of antipodal points does not reduce the intrinsic dimensionality of the parameter space. Formally, the quotient space $\mathbb{R}^n/\sim$ (where $x \sim -x$) remains n-dimensional, so $\ell = n$ suffices to represent all distinguishable aberration states. In practice, one might choose $\ell > n$ to allow

the capturing of additional variations in the data, but in our setting we were able to mitigate such variations through careful data generation and preprocessing.

The parameters of the VAE α and β are optimized to maximize the evidence lower bound (ELBO)

$$\mathcal{L} = \mathbb{E}_{q_\alpha(z|\tilde{y})}[\log p_\beta(\tilde{y}|z)] - \mathrm{KL}(q_\alpha(z|\tilde{y})\|p(z)), \tag{3.7}$$

where $p(z) = \mathcal{N}(0, I_\ell)$ is the prior distribution and KL denotes Kullback-Leibler divergence. The first term encourages accurate reconstruction of the input despite the low-dimensional latent space, while the second term regularizes the latent distribution to be close to the prior, which is typically chosen as a standard normal distribution. This regularization encourages smoothness and structure in the latent space and typically improves generalization [89].

3.3.2 Latent-Space Observation Model

The result of VAE training can be seen in Figure 3.4, which shows a latent space learned from simulated Ronchigrams with two varying aberration parameters. We limit the number of varying aberrations and latent space dimensions to two for visualization purposes. By color-coding the latent representations according to aberration values, we observe that the VAE groups similar aberration values together in the latent space. Additionally, distinct aberration values that produce visually similar Ronchigrams are also mapped to similar latent representations, reinforcing the non-injectivity of the observation model. Finally, we can observe the evenness property in the latent space: points with opposite aberration values (x and $-x$) are mapped to the same latent representation.

After training, the encoder mean $\mu_\alpha(\tilde{y})$ provides the latent representation. This establishes the latent-space observation model that will be used for state estimation:

$$\begin{aligned} x(t+1) &= x(t) + u(t) \\ z(t) &= f(x(t)) + \nu(t) \end{aligned} \tag{3.8}$$

where $f : \mathbb{R}^n \to \mathbb{R}^\ell$ is the composition of the encoder mean μ_α and h from (3.3), and $\nu(t)$ denotes the stochastic variation in the latent space induced by the Poisson noise in (3.2). The latent observations $z(t) \in \mathbb{R}^\ell$ have much lower dimension than the original Ronchigrams ($\ell \ll p$).

Crucially, because the VAE is trained on data satisfying $h(-x) = h(x)$ (established in Section 3.2.2), the learned latent mapping $f = \mu_\alpha \circ h$ inherits this symmetry by composition: $f(-x) = \mu_\alpha(h(-x)) = \mu_\alpha(h(x)) = f(x)$ for all $x \in \mathbb{R}^n$.

We assume $\nu(t) \sim \mathcal{N}(0, \Sigma_\nu)$, with $\Sigma_\nu \succ 0$ diagonal, is i.i.d. Gaussian noise independent of $x(t)$. This is justified in our setting by the delta-method and central limit theorem heuristics: under mild smoothness of μ_α and when each latent

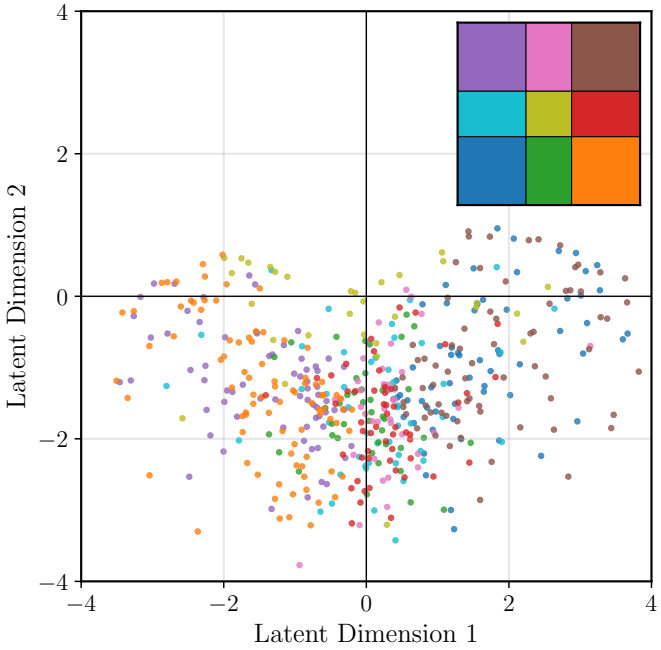


Figure 3.4. Learned VAE latent space for simulated data that contains two varying aberration parameters: defocus ($C1$), and two-fold astigmatism in the x-direction ($A1x$). The color of the data points is determined by the aberration region, with the assignment shown in the top right of the figure and in Figure 3.1. We can observe that similar aberrations are grouped together. Additionally, different aberration values that produce similar Ronchigrams are still mapped to the same latent space values (example: purple and orange), demonstrating the non-injectivity of the observation model.

coordinate aggregates many independent pixel contributions, different Poisson realizations of $y(t)$ induce approximately Gaussian fluctuations in $\mu_\alpha(\mathcal{P}\{y(t)\})$ [94–96].

With the low-dimensional latent representations $z(t)$ in place, estimating x_0 from $\{z(t)\}_{t=0}^{T}, \{u(t)\}_{t=0}^{T-1}$ becomes more tractable. If the VAE is trained on representative real data, the latent mapping f can be estimated directly from training data and standard state estimation methods (such as maximum likelihood or Bayesian filtering) can be applied to estimate x_0. This constitutes a VAE-based state estimation approach that is of independent interest. However, when the VAE is trained on simulated data, the simulation-to-reality gap means that f cannot be assumed known; it must also be estimated to account for undermodeling and

day-to-day differences in the STEM. The next section develops a joint estimation framework to simultaneously estimate both x_0 and f while exploiting the symmetry properties established in Section 3.2.2. Prior knowledge about f from the simulated dataset is leveraged through Bayesian priors to accelerate convergence, but the framework does not rely on it being exact. As a special case, this framework recovers the VAE-based state estimation approach when f is known from training data. The framework also enables adaptive input selection to accelerate convergence.

3.4 Joint State and Model Estimation Using Expectation Maximization

This section establishes that the joint estimation of x_0 and f is globally identifiable for symmetrical observation models. We then detail an Expectation-Maximization (EM) algorithm to solve the joint estimation problem.

3.4.1 Identifiability of the Joint Estimation Problem

The joint estimation of both the initial state x_0 and the latent mapping f from $\{z(t)\}_{t=0}^{T}, \{u(t)\}_{t=0}^{T-1}$ is fundamentally ambiguous due to translation symmetry and bilinear coupling. This section establishes conditions under which global identifiability is guaranteed. Throughout this section, we assume a linear basis function representation $f(x) = \Phi(x)^\top \theta$ where $\Phi(x) = [\phi_1(x), \ldots, \phi_m(x)]^\top$ is a vector of basis functions and $\theta \in \mathbb{R}^m$ are parameters. These results cover the Gaussian process model used in Section 3.4, whose posterior mean for any finite dataset is the prior mean plus a linear model of exactly this form, as detailed after Theorem 3.2.

In this setting, the joint least squares problem

$$(x_0^*, \theta^*) = \arg\min_{x_0, \theta} \sum_{t=0}^{T-1} \|z(t) - \Phi(x_0 + s(t))^\top \theta\|^2 \tag{3.9}$$

is fundamentally ambiguous due to the coupling between x_0 and θ. The following lemma formalizes this ambiguity.

Lemma 3.1. *Let $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ be a vector of basis functions, and (x_0^*, θ^*) be a solution to* (3.9)*. If*

$$\Phi(x)^\top \theta^* = \Phi(x + \delta)^\top \bar{\theta} \quad \forall x \in \mathbb{R}^n \tag{3.10}$$

holds for some nonzero shift $\delta \in \mathbb{R}^n$, $\delta \neq 0$, and vector $\bar{\theta} \in \mathbb{R}^m$, then the pair $(x_0^* + \delta, \bar{\theta})$ is also a solution, making the optimization problem (3.9) *ill-posed.*

The proof follows from direct substitution (see Appendix 3.8). The ambiguity is a translation, where a shift in the state estimate is absorbed into a reparametrization of f while the observations stay the same. No amount of data removes such an ambiguity, so identifiability must come from structural assumptions on the basis [97, 98]. The choice of the resolving structure is in general not unique. In our setting the physics supplies one: the evenness of the latent mapping in (3.5). Together with two regularity assumptions on the basis, evenness is sufficient for identifiability. We assume throughout that the applied inputs are rich enough that two mappings agreeing on the observed states agree on all of $\mathbb{R}^n$, which is the role of the input selection in Section 3.4.3.

Theorem 3.2 (Global identifiability of (x_0, θ))**.** *Let the basis functions $\{\phi_i\}_{i=1}^m$ satisfy*

(C1) each ϕ_i is even, $\phi_i(-x) = \phi_i(x)$ for all $x \in \mathbb{R}^n$;

(C2) no nonzero function in $\operatorname{span}\{\phi_i\}$ admits a nonzero period vector;

(C3) the ϕ_i are linearly independent.

If two parametrizations with $\theta \neq 0$ produce identical measurement functions,

$$\Phi(x+\xi)^\top \theta = \Phi(x+\xi')^\top \theta' \quad \forall x \in \mathbb{R}^n, \tag{3.11}$$

then $\xi = \xi'$ and $\theta = \theta'$. The pair (x_0, f) is then globally identifiable.

The proof is provided in Appendix 3.8. The mechanism behind the result is a reflection argument. Suppose two even mappings f and $\bar{f}$ in the model class satisfy $f(x) = \bar{f}(x+\delta)$ for all x. Evenness of f gives $\bar{f}(x+\delta) = \bar{f}(-x+\delta)$, so $\bar{f}$ is symmetric about δ as well as about the origin. A function with two centers of symmetry is periodic with period 2δ, which (C2) excludes for nonzero δ. This is the key step, since evenness on its own leaves the shift free and only aperiodicity removes it. With δ fixed to zero, the linear independence in (C3) recovers θ. The condition $\theta \neq 0$ is what keeps the state observable, as a zero mapping carries no information about x_0. Lemma 3.1 permits a shifted solution for every shift satisfying (3.10), in general a continuum, and the theorem excludes them all.

Remark 3.3. *The assumption that f lies exactly in the span of Φ can be relaxed. If f is well-approximated by its projection onto the basis, i.e., $\|f(x) - \Phi(x)^\top \theta^*\| \leq \epsilon$ uniformly for some $\epsilon > 0$, then conditions (C1)–(C3) applied to the projection still identify x_0 and θ. The translation ambiguity of Lemma 3.1 is present whether or not f lies exactly in the span, so the same structural conditions are what resolve it. The state estimate then degrades gracefully with ϵ, though the size of the error also depends on how sensitively the observations vary with x_0 near the true state.*

Theorem 3.2 concerns a fixed, finite basis, while the next section models f with a Gaussian process. The form of the GP estimate reconciles the two settings. For any finite dataset, the GP posterior mean is the prior mean plus a linear combination of kernel functions centered at the observed states, and this posterior mean is the estimate of f that the EM algorithm fits for each candidate state. Conditions (C1)–(C3) therefore need to hold for these estimates, and Appendix 3.8 verifies them for the symmetrized kernel introduced there. The evenness and aperiodicity built into that kernel are what separate the true state from its shifted alternatives.

3.4.2 EM Algorithm for Joint State and Model Estimation

Under conditions (C1)–(C3) established in Theorem 3.2, we can jointly estimate the initial aberration state x_0 and the latent mapping f. We do so using an Expectation-Maximization (EM) approach. We model f using a Gaussian process (GP) to generate flexible, non-parametric representations that also provide uncertainty quantification. We discretize the initial state space into a finite set of candidate states to enable tractable computation of the E-step. Details on the selection and refinement of these candidate states are provided in Section 3.4.3.

Unlike standard EM applications where the M-step (model parameter estimation) is the primary goal, our setting emphasizes the E-step, as it directly yields the aberration state estimate x_0. This focus on state estimation is motivated by the calibration objective and enabled by the identifiability results from Section 3.4.1.

Gaussian Process Representation

We model the latent mapping f using a Gaussian process (GP) prior. This modeling choice provides flexibility and computational tractability. The GP framework can approximate a broad class of functions arbitrarily well with appropriate kernel selection [78] and allows us to quantify uncertainty in the learned mapping. While our choice of the squared exponential kernel (detailed below) imposes smoothness assumptions on f, this is physically motivated: we expect the aberration-to-latent mapping to vary continuously with aberration parameters. The framework readily accommodates alternative kernels (e.g., Matérn, rational quadratic) if different smoothness or regularity properties are desired.

We assume a Gaussian process prior on f:

$$f \sim \mathcal{GP}(\mu_f, k(\cdot, \cdot)), \tag{3.12}$$

with mean function $\mu_f : \mathbb{R}^n \to \mathbb{R}$ and covariance (kernel) function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. For notational simplicity, we consider the scalar case ($\ell = 1$) in the equations presented in this section. The extension to multi-dimensional latent spaces ($\ell > 1$) is straightforward. Each output dimension can be modeled independently using a

separate GP with the same kernel structure. That is, $f(x) = [f_1(x), \ldots, f_\ell(x)]^\top$ where each $f_j \sim \mathcal{GP}(\mu_{f,j}, k(\cdot,\cdot))$. For a given dataset, this GP regression is regularized least squares with the kernel generating the basis, so the parametric model of Section 3.4.1 and the GP model are two descriptions of the same estimator [99, Section 3].

To enforce the symmetry condition (C1), we symmetrize a base kernel k_0 over the reflection group $\{I, -I\}$ acting on both arguments [100],

$$k(x, x') = \frac{1}{4}\left(k_0(x, x') + k_0(-x, x') + k_0(x, -x') + k_0(-x, -x')\right). \tag{3.13}$$

This average is a valid covariance function for any base kernel k_0, and each of its sections $k(\cdot, x')$ is even, so every posterior mean and every sample path of $\mathcal{GP}(\mu_f, k)$ is even whenever μ_f is even. For a stationary and reflection-symmetric base kernel, such as the multivariate squared exponential

$$k_0(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2}(x - x')^\top L^{-1}(x - x')\right), \tag{3.14}$$

with signal variance $\sigma_f^2 > 0$ and lengthscale matrix $L \succ 0$, the four terms collapse pairwise and (3.13) reduces to a two-term form $k(x, x') = \frac{1}{2}(k_0(x, x') + k_0(x, -x'))$. The satisfaction of conditions (C1)–(C3) for the resulting model is verified in Appendix 3.8.

In practice, the prior mean μ_f and kernel hyperparameters (such as Σ_ν, σ_f^2 and L) can be estimated from the large simulated dataset using standard maximum likelihood or cross-validation methods [78]. This allows us to incorporate prior knowledge from physics-based simulations while retaining the flexibility to adapt to day-to-day variations by fitting the GP posterior to the current observations.

Candidate Initial States and Likelihood

To estimate x_0 using the EM algorithm, we discretize the initial state space into a finite set of candidate states. This discretization enables tractable computation of the E-step by evaluating the likelihood of each candidate under the GP model. Consider a discrete set of candidate initial states $\{x_0^{(i)}\}_{i=1}^N$. The selection of the candidate set is discussed in Section 3.4.3. We place a uniform prior $p(x_0^{(i)}) = 1/N$ on the candidates. Given candidate $x_0^{(i)}$ and the deterministic dynamics (3.1), the aberration trajectory is $x(t) = x_0^{(i)} + s(t)$ for $t = 0, 1, \ldots, T$.

Under the GP prior (3.12) and the observation model (3.8) conditioned on candidate $x_0^{(i)}$, the latent observations have marginal distribution (for the scalar case $\ell = 1$):

$$p(Z|x_0^{(i)}) = \mathcal{N}(Z; \mu_Z^{(i)}, K^{(i)} + \Sigma_\nu I_{T+1}), \tag{3.15}$$

where

$$Z = \big[z(0), z(1), \ldots, z(T)\big]^\top \in \mathbb{R}^{T+1} \tag{3.16}$$

is the stacked latent observation vector over the time horizon. The stacked prior mean vector $\mu_Z^{(i)} \in \mathbb{R}^{T+1}$ evaluates the prior mean function μ_f at the aberration trajectory implied by candidate i:

$$\mu_Z^{(i)} = \begin{bmatrix} \mu_f(x_0^{(i)} + s(0)) \\ \mu_f(x_0^{(i)} + s(1)) \\ \vdots \\ \mu_f(x_0^{(i)} + s(T)) \end{bmatrix}. \tag{3.17}$$

The prior covariance matrix $K^{(i)} \in \mathbb{R}^{(T+1)\times(T+1)}$ has elements

$$K_{tr}^{(i)} = k(x_0^{(i)} + s(t), x_0^{(i)} + s(r)), \quad t, r = 0, 1, \ldots, T. \tag{3.18}$$

Finally, $\Sigma_\nu > 0$ is a scalar (in the $\ell = 1$ case) representing the variance of the latent observation noise $\nu(t)$ from (3.8). The total covariance in (3.15) combines the GP prior covariance $K^{(i)}$ with the observation noise $\Sigma_\nu I_{T+1}$. For $\ell > 1$, we stack all latent dimensions and extend the covariance structure using Kronecker products. All subsequent expressions extend naturally with this block structure. With these definitions, we can evaluate the GP marginal likelihood $p(Z|x_0^{(i)})$ for each candidate, which forms the basis for the EM algorithm detailed next.

Evidence Lower Bound and EM Iterations

The EM algorithm finds f and x_0 by maximizing the Evidence Lower Bound (ELBO) and alternating between estimating the posterior distribution over x_0 (E-step) and updating the (GP) representation of f (M-step) [90]. Let $\pi : \{x_0^{(1)}, \ldots, x_0^{(N)}\} \to [0, 1]$ be a probability distribution over the candidate set, where $\pi(x_0^{(i)})$ denotes the probability assigned to candidate i and $\sum_{i=1}^N \pi(x_0^{(i)}) = 1$. The ELBO as a function of f and π is given by

$$\mathrm{ELBO}(\pi, f) = \sum_{i=1}^{N} \pi(x_0^{(i)}) \log p(Z|x_0^{(i)}, f) - \mathrm{KL}(\pi(x_0)\|p(x_0)), \tag{3.19}$$

where $\mathrm{KL}(\pi\|p) = \sum_{i=1}^N \pi(x_0^{(i)}) \log \frac{\pi(x_0^{(i)})}{p(x_0^{(i)})}$ is the standard discrete KL divergence. The two steps of the EM algorithm are detailed next.

M-step: Update the GP representation of f given the observations Z and the current distribution π over candidates. Since the ELBO's dependence on f

enters through the likelihoods $\log p(Z|x_0^{(i)}, f)$, and these are maximized by the GP posterior conditioned on each candidate, we compute the GP posterior for each candidate i. For each candidate $x_0^{(i)}$, the GP posterior mean conditioned on that candidate and on the observations Z is given by

$$\mu_{f|Z}^{(i)}(x) = \mu_f(x) + k_x^{(i)}(K^{(i)} + \Sigma_\nu I_{T+1})^{-1}(Z - \mu_Z^{(i)}), \tag{3.20}$$

where $k_x^{(i)} = [k(x, x_0^{(i)} + s(0)), \ldots, k(x, x_0^{(i)} + s(T))]$ is the kernel slice evaluated at the query point x and the trajectory points for candidate i. We can then form the overall posterior of f as a mixture of these individual posteriors weighted by the distribution π over candidates. We obtain

$$\mu_{f|Z}(x) = \sum_{i=1}^{N} \pi(x_0^{(i)}) \mu_{f|Z}^{(i)}(x), \tag{3.21}$$

assuming $\ell = 1$. For $\ell > 1$, the expressions extend in a straightforward manner with the block structure.

E-step: Update the distribution π over candidates. Maximizing the ELBO over π subject to $\sum_{i=1}^{N} \pi(x_0^{(i)}) = 1$ yields $\pi(x_0^{(i)}) \propto p(x_0^{(i)}) p(Z|x_0^{(i)})$. Rather than conditioning on the current f, we evaluate the marginal likelihood (3.15), in which f is integrated out under its GP prior. This accounts for the remaining model uncertainty and makes the weights independent of the M-step, as discussed below. Assuming a uniform prior over candidates $p(x_0^{(i)}) = 1/N$, we obtain the posterior π by evaluating the likelihoods $p(Z|x_0^{(i)})$ for each candidate and normalizing. The log-likelihood of observing Z given candidate i is

$$\begin{aligned} \log p(Z|x_0^{(i)}) = &-\frac{1}{2}(Z - \mu_Z^{(i)})^\top (K^{(i)} + \Sigma_\nu I_{T+1})^{-1}(Z - \mu_Z^{(i)}) \\ &- \frac{1}{2}\log |K^{(i)} + \Sigma_\nu I_{T+1}| - \frac{T+1}{2}\log(2\pi). \end{aligned} \tag{3.22}$$

We then compute the posterior weights as $w_i \propto p(Z|x_0^{(i)})$ and normalize such that $\sum_{i=1}^{N} w_i = 1$ to obtain the discrete distribution $\pi(x_0^{(i)}) = w_i$. For $\ell > 1$, the expressions extend with the block structure detailed earlier.

With f integrated out, the E-step maximizes the evidence lower bound on $\log p(Z)$ exactly: the resulting weights are the posterior $p(x_0^{(i)} \mid Z)$, at which the bound is tight [90, 101]. Theorem 3.2 establishes that under conditions (C1)–(C3), the pair (x_0, f) is globally identifiable, meaning there exists a unique solution to the joint estimation problem. However, the identifiability result does not guarantee that the EM algorithm will converge to the global optimum. Establishing global convergence remains an open theoretical question, though empirically the algorithm converges reliably to accurate estimates in our experiments.

An important observation about our particular setting is that the E-step does not depend on the full GP mixture posterior from the M-step, but only on the individual candidate likelihoods computed from the individual GP posteriors. With a uniform prior, the E-step weights remain constant across EM iterations for a fixed dataset. This means we can compute the E-step once per dataset, after which a single M-step produces the final estimates. In practice, we often run only a single EM iteration per dataset, significantly reducing computational cost.

An alternative to the EM approach presented above is *hard EM* that replaces the full posterior distribution $\pi(x_0)$ with a point estimate at each iteration. Specifically, in the hard E-step, instead of computing the full distribution over candidates, we select $\hat{x}_0^{(i^*)}$ where $i^* = \arg\max_i \bar{w}_i$, effectively setting $\pi(x_0^{(i^*)}) = 1$ and $\pi(x_0^{(j)}) = 0$ for $j \neq i^*$. This simplifies the M-step: the mixture of GP posteriors (3.21) reduces to a single GP posterior conditioned on the most likely candidate, and eliminates the need to track and update weights for all candidates. As a consequence, the computational complexity of the model estimation step is reduced from $O(N \cdot T^3)$ to $O(T^3)$ per iteration. While hard EM is computationally more efficient and requires less memory (no mixture representation), it is more prone to getting stuck in local optima. This typically happens because it commits to a single candidate at each iteration rather than maintaining uncertainty over multiple plausible candidates. The full EM approach is generally preferred when computational resources permit, as it better explores the posterior landscape. In our experiments, we employ the full EM algorithm as the computational cost remains manageable due to the moderate number of candidates used.

3.4.3 Implementation Details

The EM algorithm detailed above can be integrated into an online calibration procedure. At each time step T, we apply a known input $u(T-1)$ and acquire a new Ronchigram observation $y(T)$. We preprocess the observation to obtain the latent representation $z(T) = \mu_\alpha(\mathcal{P}\{y(T)\})$. We then augment the dataset with the new latent space observation and input, updating the cumulative input $s(T) = \sum_{t=0}^{T-1} u(t)$. With the updated dataset $\{z(t), s(t)\}_{t=0}^{T}$, we run the EM iterations to refine the estimates of x_0 and f.

There are several reasons to acquire observations one at a time and to re-run EM with every new observation. First, it allows us to adaptively select the input $u(T)$ based on current estimates, improving convergence speed. Second, it allows us to select new candidate initial states based on the current posterior, reducing the total number of candidates while maintaining accuracy. Finally, it allows us to monitor convergence and stop the calibration process once satisfactory estimates are obtained, reducing unnecessary data collection.

Input Selection Strategy

The input selection strategy can significantly impact convergence speed and calibration accuracy. While random inputs can provide sufficient excitation for estimation, more sophisticated strategies can select inputs that maximize information gain about the unknown parameters. A simple acquisition heuristic is to select inputs that yield measurements at aberration locations where the GP posterior variance is highest, thereby reducing uncertainty in the latent mapping f. For computational efficiency, we employ a *hard EM*-inspired approximation: rather than computing the full mixture posterior variance, we evaluate the variance using only the most likely candidate $i^* = \arg\max_i w_i$. The posterior variance for this candidate is

$$\mathrm{Var}_{f|Z}^{(i^*)}(x) = k(x,x) - k_x^{(i^*)\top}(K^{(i^*)} + \Sigma_\nu I_{T+1})^{-1} k_x^{(i^*)}, \tag{3.23}$$

where $k_x^{(i^*)}$ is the cross-covariance vector from (3.20). This approximation reduces the computational cost from $O(N \cdot T^3)$ to $O(T^3)$ per variance query, making variance-based input selection practical. The approximation is justified when one candidate dominates the posterior (large $\max_i w_i$), as typically occurs after a few EM iterations. Inputs are then selected through

$$u(T) = \arg\max_{u \in \mathcal{U}} \mathrm{Var}_{f|Z}^{(i^*)}(x_0^{(i^*)} + s(T) + u), \tag{3.24}$$

where $\mathcal{U}$ is the set of allowable inputs (e.g., bounded defocus and astigmatism values). In practice, we optimize (3.24) using grid search over a randomly sampled set of candidate inputs from $\mathcal{U}$.

Candidate Refinement

In between acquiring new observations, we can refine the candidate set $\{x_0^{(i)}\}_{i=1}^N$ based on previous EM estimates to focus computational resources on high-probability regions while maintaining exploration. This refinement strategy is conceptually similar to resampling in particle filtering [92], where particles are redistributed based on their importance weights to concentrate samples in high-probability regions while avoiding sample degeneracy.

Our refinement strategy balances three objectives: exploiting the current posterior by retaining high-weight candidates, refining the search locally around promising regions, and maintaining global exploration to avoid premature convergence. Let w_i denote the posterior weights from the most recent E-step. We construct the new candidate set by combining three components. First, we keep a fraction of the top-weighted candidates to preserve the current best estimates. Second, we generate new candidates by perturbing high-weight candidates with Gaussian noise, sampling parent candidates proportionally to their weights:

$$x_0^{\text{new}} = x_0^{\text{parent}} + \mathcal{N}(0, \sigma^2 I_n), \tag{3.25}$$

Algorithm 1 Iterative VAE-EM Calibration Algorithm

Input: Trained VAE encoder μ_α, candidate set $\{x_0^{(i)}\}_{i=1}^N$, GP prior mean μ_f and kernel $k(\cdot,\cdot)$
Output: Estimated initial state $\hat{x}_0$, latent mapping $\hat{f}$

Initialize $T = 0$, $s(0) = 0$
Acquire initial observation $y(0)$, compute $\tilde{y}(0) = \mathcal{P}\{y(0)\}$, encode $z(0) = \mu_\alpha(\tilde{y}(0))$
repeat
    // EM iterations on dataset $\{z(t), s(t)\}_{t=0}^T$
    repeat
        E-step: Compute weights w_i via (3.22) with $Z = [z(0), \ldots, z(T)]^\top$
        M-step: Update $\mu_{f|Z}(x)$ through (3.21)
    until EM convergence
    update candidate set $\{x_0^{(i)}\}_{i=1}^N$ using (3.25)
    Select next input $u(T)$ using (3.24)
    $s(T+1) \leftarrow s(T) + u(T)$, $T \leftarrow T+1$
    Acquire $y(T)$, compute $\tilde{y}(T) = \mathcal{P}\{y(T)\}$, encode $z(T) = \mu_\alpha(\tilde{y}(T))$
until convergence criterion met

where the parent is drawn with probability proportional to w_i and $\sigma > 0$ controls the exploration-exploitation tradeoff. Finally, we sample a small fraction of candidates uniformly over the aberration space to maintain coverage and prevent convergence to local optima. After refinement, the candidate weights are reset uniformly and the GP posteriors are recomputed in the next M-step.

3.4.4 Algorithm Summary

The complete calibration procedure is summarized in Algorithm 1. The VAE is trained offline on a dataset of simulated Ronchigram images. During online STEM calibration, the algorithm iteratively collects new Ronchigram observations, encodes them, and refines both the aberration estimate x_0 and the latent mapping f using EM iterations.

3.5 Results

We validate the proposed VAE-EM calibration method by applying it to real STEM data acquired on different days to capture day-to-day variations in system behavior. We compare the proposed method with existing automated calibration approaches in terms of calibration accuracy and convergence speed. In our previous work, we

presented a detailed comparison of different Ronchigram-based calibration methods from the literature in terms of accuracy and speed. We refer the reader to that work for a comprehensive overview of existing approaches. Here, we focus on comparing our proposed VAE-EM method against the strongest baselines identified previously.

Our experiments consider three aberration parameters ($n = 3$): defocus ($C1$) and two-fold astigmatism in both x- and y-directions ($A1x$, $A1y$). We use latent dimension $\ell = n = 3$ for the EM estimation experiments. For visualizations of latent spaces, we show results with $\ell = n = 2$ (defocus and x-astigmatism only).

3.5.1 VAE Captures Aberration Structure in Latent Space

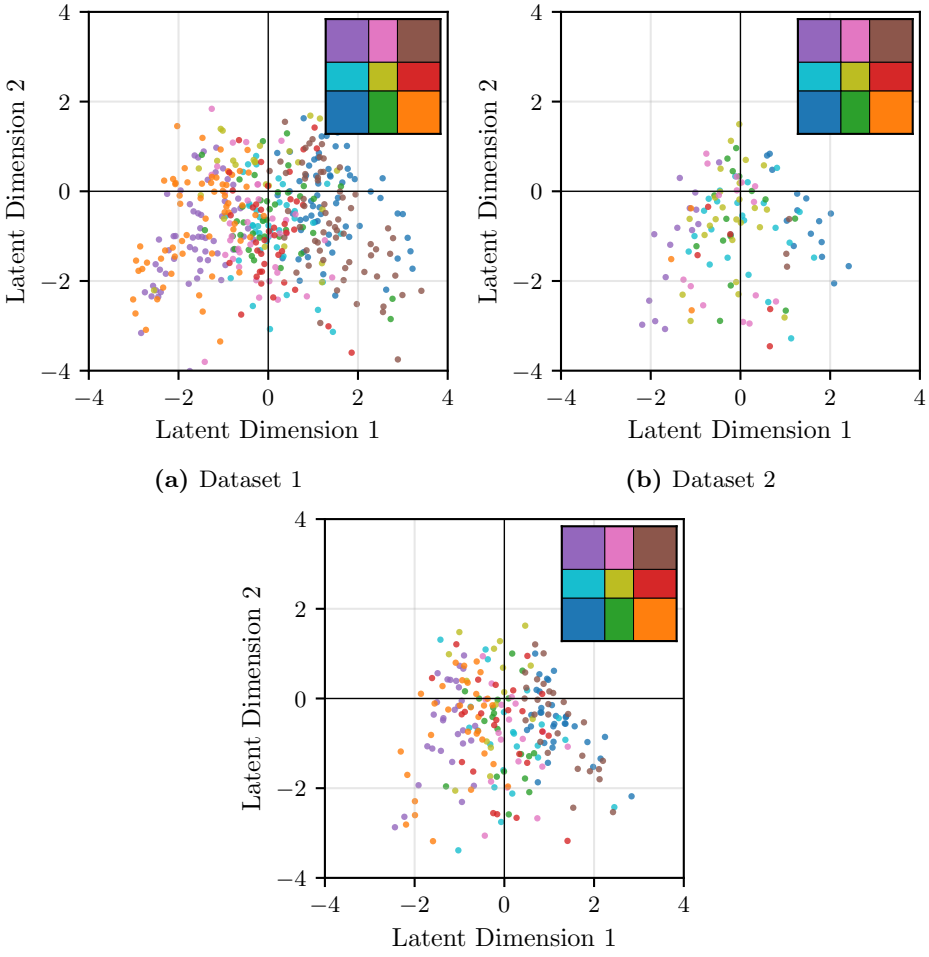
Figure 3.5 shows the latent space distribution of the real STEM Ronchigram datasets using a 2D latent space for visualization.

We see consistency across the different datasets, as well as with respect to the simulated data from Figure 3.4. Observe that the experimental results are somewhat noisier and occupy a smaller region of the latent space compared to the simulated data. This is likely due to measurement noise and day-to-day variations in the microscope that are not captured by the simulator. Nevertheless, the overall structure is preserved: aberration states x and $-x$ continue to map to similar latent locations (as seen by the color coding), while separation between different aberration quadrants is maintained. This indicates that the VAE has learned a meaningful representation that reflects the underlying physics.

From the latent space representation, we can fit the GP prior mean function $\mu_f(x)$ that maps aberration states to latent space dimensions. We use a grid of symmetrical piecewise linear bases, though other basis functions (e.g., radial basis functions, polynomials) could also be used. Figure 3.6 shows the fitted prior mean for the 2D latent space case. We see smooth mappings that capture the nonlinear relationship between aberration states and latent dimensions. These fitted prior means typically improve estimation accuracy for low numbers of observations, though we see limited improvements once enough data has been acquired.

3.5.2 Superior Calibration Performance on Real STEM Data

Figure 3.7 and Figure 3.8 show the aberration estimation error 2-norm as a function of the number of Ronchigram observations used for calibration, averaged over 100 Monte Carlo runs with different random initializations. After approximately 10 observations, the proposed VAE-EM method achieves the accuracy requirements for high-quality imaging (defocus and astigmatism below 10 nm each) on over 50% of runs. On all datasets, the proposed method converges after around 20 observations to very accurate estimates with high consistency across random initializations. The resulting calibration error 2-norms are reduced by over a factor of 2 compared to our previous Bayesian optimization approach [62] on the same

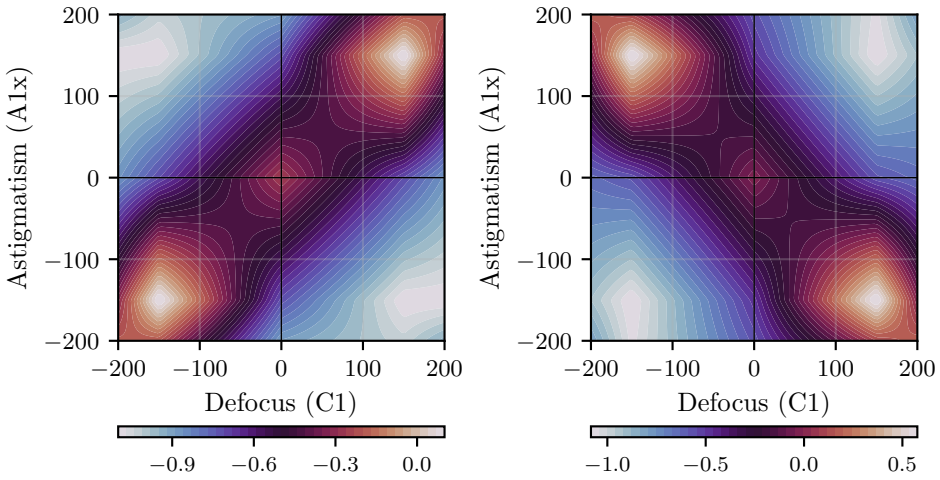


(a) Dataset 1 **(b)** Dataset 2

(c) Dataset 3

Figure 3.5. Latent space distribution for Ronchigram datasets gathered on real STEM on different days. The latent space data points are colored according to their aberration values, with the color-coding scheme taken from Figure 3.1. We can observe consistency across datasets, as well as with respect to the simulated data from Figure 3.4, indicating that the VAE has learned a meaningful latent representation that reflects the underlying aberration states.

datasets. Our previous work established that this Bayesian optimization approach already matched the performance of existing automated methods on real STEM



(a) First latent dimension z_1 **(b)** Second latent dimension z_2

Figure 3.6. Fitted prior mean function $\mu_f(x)$ mapping aberration parameters to VAE latent space, shown as a function of defocus ($C1$) and two-fold astigmatism ($A1x$). The prior mean is generated by fitting piecewise linear bases to a large set of simulated Ronchigrams. Note the symmetry with respect to the origin, reflecting the physical property $f(-x) = f(x)$.

data. Current commercial solutions such as CEOS and OptiSTEM(+) typically require approximately one minute to reach acceptable calibration specifications. In contrast, our VAE-EM method requires only around 20 Ronchigrams, corresponding to roughly 15 seconds of acquisition time, while achieving substantially higher accuracy.

3.6 Discussion

We have presented a VAE-EM approach for automated STEM calibration that addresses two fundamental limitations of existing data-driven methods: the reliance on scalar image representations and the simulation-to-reality gap. Multi-dimensional VAE representations preserve richer information about the aberration state than scalar quality metrics, which rapidly lose information content as the number of calibration parameters increases. By jointly estimating both the calibration state and the latent mapping using EM, we achieve over 2× more accurate calibration compared to state-of-the-art methods while requiring fewer observations.

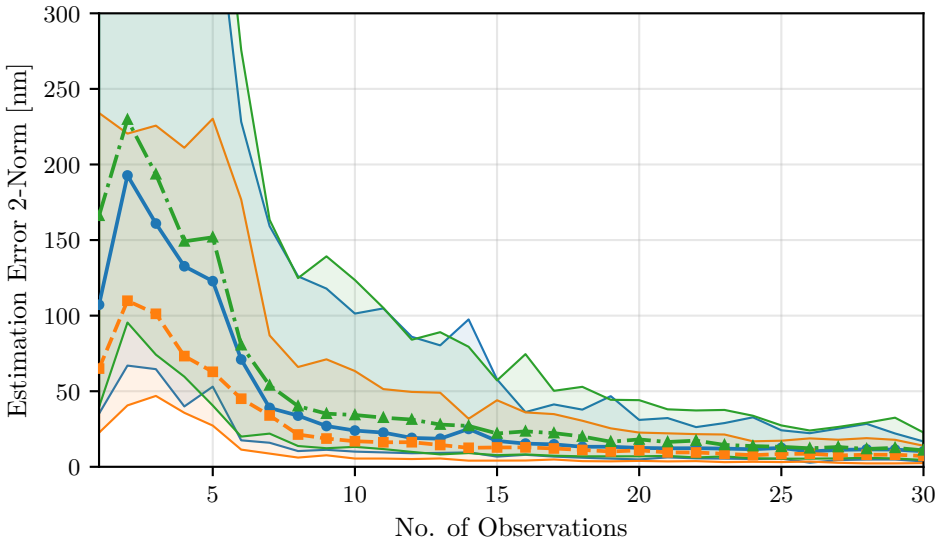
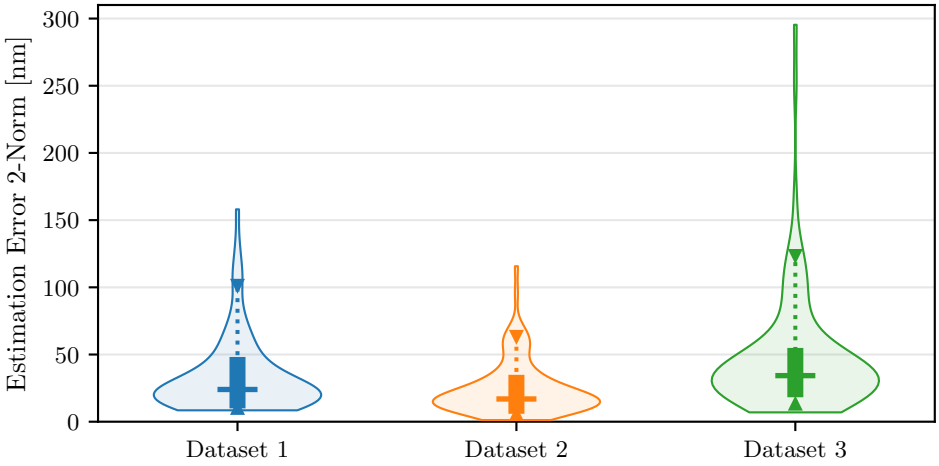


Figure 3.7. Estimation performance on real STEM data over 100 Monte Carlo runs with different random initializations with 3 aberration parameters $(C1, A1x, A1y)$. We view the aberration estimation error 2-norm as a function of the number of Ronchigram observations used for calibration for Dataset 1 (—●—), Dataset 2 (- ■ -), and Dataset 3 (- ▲ -). Plotted are the medians over the 100 runs, with the shaded bands spanning the 5th to 95th percentile. The proposed VAE-EM method consistently outperforms existing automated calibration approaches in terms of speed and accuracy.

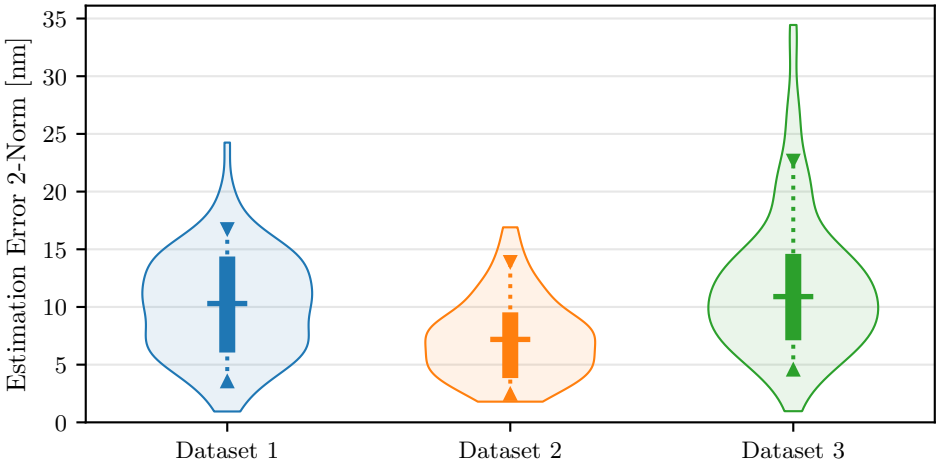
The global identifiability result (Theorem 3.2) provides theoretical grounding for our approach by ensuring that the joint estimation problem has a unique solution under mild conditions on the basis functions. This is notable because standard EM approaches for inverse problems often lack such guarantees [31, 91]. Our approach provides an elegant way of selecting these basis functions: they are generated by a symmetrized kernel, which in the limit can approximate any even function arbitrarily well. Prior knowledge can be embedded through the choice of base kernel, prior mean, and hyperparameters.

3.6.1 Limitations and Future Work

A limitation of our approach is the need for substantial training data to train the VAE. We resolve this by using a digital twin (simulator) and explicitly learning the resulting model mismatch through the GP posterior. However, the simulated dataset



(a) 10 observations



(b) 30 observations

Figure 3.8. Estimation error performance on real STEM data after 10 and 30 Ronchigram observations. The violin plots summarize the distribution of estimation error 2-norms with 3 aberration parameters ($C1, A1x, A1y$) across 100 Monte Carlo runs with different random initializations, for dataset 1 (■), dataset 2 (■), and dataset 3 (■). Within each violin, the horizontal dash marks the median and the thick bar the interquartile range. The dotted line spans the 5th to 95th percentile, bounded by the triangles.

must cover essentially all aberration combinations that we aim to estimate in order to train a sufficiently expressive VAE. For practical deployment, we recommend using iterative candidate refinement and variance-based input selection. These strategies significantly reduce computational cost while maintaining calibration accuracy. If further reduction is needed, hard EM can be employed at the cost of some estimation accuracy.

Generalization to higher-order aberrations is conceptually straightforward since the evenness constraint still holds, though practical challenges arise from GP computation scaling and the curse of dimensionality. A hierarchical approach that first calibrates lower-order aberrations before refining higher-order terms may address these challenges. Additionally, higher-order aberrations tend to have localized effects on Ronchigrams, which may require patching the Ronchigrams and applying Fourier transforms locally to extract relevant latent features.

3.6.2 Applicability Beyond STEM

While developed for STEM calibration, the methods presented here may extend to other inverse problems. We now outline when the framework applies.

The starting point is an inverse problem where high-dimensional observations arise from low-dimensional parameters. In our case, Ronchigrams contain approximately 10^5 pixels yet are determined by a few aberration parameters. This gap enables a VAE to learn a compressed representation without losing estimation-relevant information. When training data comes from a simulator rather than the real system, the learned latent mapping will not perfectly match reality. This simulation-to-reality gap means the VAE representations are biased. Joint estimation of both the state and the residual model mismatch using EM can correct this bias, but such joint estimation is generally ill-posed: many combinations of state and model parameters may explain the observations equally well.

Structural constraints can restore identifiability. In our case, the evenness constraint from Fourier optics allows us to construct a symmetry-constrained GP kernel, which establishes global identifiability (Theorem 3.2). Non-injectivity of the observation mapping is not itself a requirement. Rather, the particular structure of our non-injectivity enables us to recover identifiability. An injective observation mapping would trivially ensure identifiability and also fit within this framework. The framework also requires that the state dynamics equations are known and invertible. Our state-space model has trivial dynamics where the next state equals the current state plus a known input, allowing all observations to be related back to a common initial state for joint estimation.

When the VAE is trained on representative real data, the simulation-to-reality gap becomes negligible. The proposed framework then reduces to pure state estimation in the learned latent space, with the latent mapping f estimated

directly from training data rather than learned online. In essence, our framework encompasses both joint state-model estimation and pure VAE-based state estimation as a special case.

Several application domains may fit the VAE-EM framework. Optical systems with known symmetries, such as adaptive optics for telescopes or microscopy, share the evenness structure that enables our identifiability result. Phase retrieval problems exhibit similar non-injectivity due to the loss of phase information. Medical imaging modalities where simulators exist but patient-specific variations cause model mismatch, such as MRI or CT reconstruction, may also benefit from joint state-model estimation. For applications where representative real data is available, such as industrial quality control with abundant labeled samples, the simpler VAE-based state estimation applies directly without requiring the EM iterations for model learning.

3.6.3 Conclusion

This work introduced a VAE-EM framework that surpasses state-of-the-art STEM calibration performance by combining multi-dimensional image representations with joint state-model estimation. The key theoretical insight is that structural constraints, such as the evenness property from Fourier optics, can restore global identifiability in joint estimation problems that are otherwise ill-posed. This provides a principled approach to bridging simulation-to-reality gaps in data-driven inverse problems.

3.7 Appendix: Optics Derivation and Symmetry of Ronchigrams

This appendix derives the symmetry property $h(-x) = h(x)$ stated in Section 3.2.2 from first principles of electron optics, largely following Kirkland [93].

3.7.1 Image Formation Model

In scanning transmission electron microscopy, aberrations modify the phase of the electron wave. In STEM mode they can be applied at the focal plane (pupil plane) of the probe forming system. Let $x \in \mathbb{R}^n$ denote the aberration state (a vector of real coefficients for defocus, astigmatism, coma, etc.). The wave aberration function is parametrized as

$$\chi(\mathbf{k}; x) = \sum_{j=1}^{n} x_j \phi_j(\mathbf{k}), \tag{3.26}$$

where $\mathbf{k} = (k_x, k_y) \in \mathbb{R}^2$ are spatial frequencies in the pupil plane and $\{\phi_j(\mathbf{k})\}_{j=1}^n$ are real-valued pupil basis functions (e.g., Zernike or Seidel polynomials). Combined with a real, symmetric aperture function $P(\mathbf{k})$, the wave at pupil plane is

$$A_x(\mathbf{k}) = P(\mathbf{k})e^{i\chi(\mathbf{k};x)}. \tag{3.27}$$

The incident probe wavefunction in real space is obtained via inverse Fourier transform

$$\psi_x(\mathbf{r}) = (\mathcal{FT}^{2D})^{-1}\{A_x\}(\mathbf{r}) := \int_{\mathbb{R}^2} A_x(\mathbf{k})e^{2\pi i\mathbf{k}\cdot\mathbf{r}}d\mathbf{k}, \tag{3.28}$$

where $\mathbf{r} = (r_x, r_y) \in \mathbb{R}^2$ are real-space coordinates and $(\mathcal{FT}^{2D})^{-1}$ denotes the inverse continuous 2D Fourier transform.

For a thin specimen, we employ the phase object approximation. The specimen transmission function is

$$T(\mathbf{r}) = e^{i\sigma V(\mathbf{r})}, \tag{3.29}$$

where $V(\mathbf{r})$ is the projected electrostatic potential and σ is the electron interaction constant (which depends on the accelerating voltage). The exit wavefunction $\psi_E(\mathbf{r};x)$ immediately after the specimen is then given by the multiplicative approximation:

$$\psi_E(\mathbf{r};x) = \psi_x(\mathbf{r})T(\mathbf{r}). \tag{3.30}$$

The Ronchigram intensity $I_x(\mathbf{k})$ is the squared modulus of the Fourier transform of the exit wave (by Fraunhofer diffraction):

$$I_x(\mathbf{k}) = \left|\mathcal{FT}^{2D}\{\psi_E\}(\mathbf{k})\right|^2, \tag{3.31}$$

where $\mathcal{FT}^{2D}\{\psi_E\}(\mathbf{k}) := \int_{\mathbb{R}^2} \psi_E(\mathbf{r})e^{-2\pi i\mathbf{k}\cdot\mathbf{r}}d\mathbf{r}$ denotes the continuous 2D Fourier transform of ψ_E. This intensity distribution corresponds to the continuous Ronchigram. Sampling it on a discrete pixel grid yields the observation $y(t)$ in the main text. The preprocessing described in Section 3.2.2 then computes the 2D Fourier transform of the Ronchigram, then takes the squared modulus. By the autocorrelation theorem [102], the Fourier transform of $I_x(\mathbf{k})$ equals the autocorrelation of the exit wave. We obtain

$$H_x(\boldsymbol{\omega}) := \mathcal{FT}^{2D}\{I_x\}(\boldsymbol{\omega}) = \int_{\mathbb{R}^2} \psi_E(\mathbf{r}+\boldsymbol{\omega};x)\psi_E^*(\mathbf{r};x)\,d\mathbf{r}, \tag{3.32}$$

with $\boldsymbol{\omega} = (\omega_x, \omega_y) \in \mathbb{R}^2$ the spatial shift coordinates (the Fourier conjugate variable to $\mathbf{k}$). Substituting the exit wave definition (3.30), we obtain

$$H_x(\boldsymbol{\omega}) = \int_{\mathbb{R}^2} [\psi_x(\mathbf{r}+\boldsymbol{\omega})\psi_x^*(\mathbf{r})] \cdot [T(\mathbf{r}+\boldsymbol{\omega})T^*(\mathbf{r})]\,d\mathbf{r}. \tag{3.33}$$

The continuous power spectrum of the Ronchigram is then

$$\mathrm{PS}_x(\boldsymbol{\omega}) = |H_x(\boldsymbol{\omega})|^2 = \left|\mathcal{FT}^{2D}\{I_x\}(\boldsymbol{\omega})\right|^2. \tag{3.34}$$

3.7.2 Symmetry Under Sign Inversion of Aberrations

Consider inverting all aberration coefficients: $x \mapsto -x$. This yields $\chi(\mathbf{k}; -x) = -\chi(\mathbf{k}; x)$, so $A_{-x}(\mathbf{k}) = [A_x(\mathbf{k})]^*$. Consequently, the probe wavefunction transforms as:

$$\psi_{-x}(\mathbf{r}) = (\mathcal{F}\mathcal{T}^{2D})^{-1}\{A_x^*\}(\mathbf{r}) = \psi_x^*(-\mathbf{r}). \tag{3.35}$$

Substituting into the autocorrelation integral (3.32):

$$H_{-x}(\boldsymbol{\omega}) = \int_{\mathbb{R}^2} \psi_x^*(-\mathbf{r} - \boldsymbol{\omega})\psi_x(-\mathbf{r})T(\mathbf{r} + \boldsymbol{\omega})T^*(\mathbf{r})\, d\mathbf{r}. \tag{3.36}$$

Changing variables $\mathbf{r}' = -\mathbf{r} - \boldsymbol{\omega}$, we have $d\mathbf{r} = d\mathbf{r}'$, and obtain

$$H_{-x}(\boldsymbol{\omega}) = \int_{\mathbb{R}^2} [\psi_x^*(\mathbf{r}')\psi_x(\mathbf{r}' + \boldsymbol{\omega})]\, [T(-\mathbf{r}')T^*(-\mathbf{r}' - \boldsymbol{\omega})]\, d\mathbf{r}'. \tag{3.37}$$

The probe term in the first bracket is identical to that in $H_x(\boldsymbol{\omega})$. The symmetry of the final result therefore depends on the autocorrelation of the specimen transmission function in the second bracket.

Comparing the second bracket with its counterpart in $H_x(\boldsymbol{\omega})$ shows that it is the same specimen autocorrelation, evaluated at spatially inverted positions. Inverting the aberrations therefore reflects the specimen autocorrelation relative to the probe autocorrelation, so the symmetry of the power spectrum is governed by the specimen structure. In the absence of a specimen (i.e., $T(\mathbf{r}) = 1$), the second bracket is constant and we obtain $H_{-x}(\boldsymbol{\omega}) = H_x(\boldsymbol{\omega})$ exactly. This is consistent with the fact that in the absence of a specimen, the Ronchigram is simply the squared modulus of the probe wavefunction, which is invariant under spatial inversion. For statistically isotropic specimens (amorphous materials), the projected potential $V(\mathbf{r})$ is a random field with statistics invariant under spatial inversion, implying that $T(\mathbf{r}) = e^{i\sigma V(\mathbf{r})}$ has the same statistical properties as $T(-\mathbf{r}) = e^{i\sigma V(-\mathbf{r})}$. The reflected autocorrelation then has the same distribution as the original one, and the power spectrum is invariant under aberration sign inversion in expectation over the specimen, $\mathbb{E}\left[|H_{-x}(\boldsymbol{\omega})|^2\right] = \mathbb{E}\left[|H_x(\boldsymbol{\omega})|^2\right]$. When using Ronchigram images for aberration correction, typically amorphous regions of the sample are used, as the goal is to observe the effects of aberrations rather than the sample itself. As such, in practice, our method is not limited by the assumptions mentioned above.

3.7.3 Practical Considerations

Even for a single specimen realization, which need not be perfectly isotropic, the symmetry is typically observed in practice for two reasons. First, the thin specimens used here are close to weak phase objects, for which $T(\mathbf{r})$ is close to unity and

the residual asymmetry vanishes with the interaction constant σ. Second, real specimens almost always have some amorphous contamination (e.g., carbon or oxide layers), which adds an inversion-invariant component to the Ronchigram texture.

In practice, the Ronchigram is sampled on an $n_h \times n_w$ pixel grid, and the preprocessing operator $\mathcal{P}$ (defined in Section 3.2.2) applies windowing, the 2D discrete Fourier transform $\mathcal{F}^{2D}$, and computes the power spectrum. Under sufficiently fine sampling, the discrete Fourier transform approximates the continuous Fourier transform at discrete frequencies. Since the continuous power spectrum satisfies $\mathrm{PS}_{-x}(\boldsymbol{\omega}) = \mathrm{PS}_x(\boldsymbol{\omega})$, and the windowing operation is symmetric and independent of x, the discrete power spectrum inherits this symmetry:

$$h(-x) = h(x) \quad \forall x \in \mathbb{R}^n. \tag{3.38}$$

This property holds in expectation even with Poisson noise, as the expected value of the discrete Fourier transform power spectrum differs from the deterministic case only by an additive constant (the total electron count) that is symmetric in x [68].

3.8 Appendix: Global Identifiability Proofs

This appendix provides the proofs of Lemma 3.1 and Theorem 3.2, and verifies that the Gaussian process model with the symmetrized kernel satisfies conditions (C1)–(C3).

3.8.1 Proof of Lemma 3.1

Let $x(t) = x_0^* + s(t)$ denote the aberration trajectory. Direct substitution using the translation invariance property (3.10) with $x = x_0^* + s(t)$ gives

$$\sum_{t=0}^{T-1} \|z(t) - \Phi(x_0^* + \delta + s(t))^\top \bar{\theta}\|^2 = \sum_{t=0}^{T-1} \|z(t) - \Phi(x_0^* + s(t))^\top \theta^*\|^2,$$

so both pairs achieve the same objective value. □

3.8.2 Proof of Theorem 3.2

Suppose two parametrizations with $\theta \neq 0$ produce identical measurement functions,

$$\Phi(x + \xi)^\top \theta = \Phi(x + \xi')^\top \theta' \quad \forall x \in \mathbb{R}^n. \tag{3.39}$$

Write $g(x) := \Phi(x)^\top \theta$ and $\bar{g}(x) := \Phi(x)^\top \theta'$ for the two mappings, and let $\delta := \xi' - \xi$. Then (3.39) reads $g(x) = \bar{g}(x + \delta)$ for all x. By condition (C1) both g and $\bar{g}$ are even. The linear independence in condition (C3) and $\theta \neq 0$ give $g \neq 0$, and since $g(x) = \bar{g}(x + \delta)$ this also gives $\bar{g} \neq 0$.

We first show $\delta = 0$. Evenness of g gives $g(x) = g(-x) = \bar{g}(-x + \delta)$, while the equivalence gives $g(x) = \bar{g}(x + \delta)$. Hence

$$\bar{g}(x + \delta) = \bar{g}(-x + \delta) \quad \forall x \in \mathbb{R}^n.$$

Substituting $u = x + \delta$ yields $\bar{g}(u) = \bar{g}(2\delta - u)$, so $\bar{g}$ is symmetric about the point δ. Combined with its evenness about the origin, $\bar{g}$ is invariant under the composition of the two point reflections, which is the translation $u \mapsto u + 2\delta$:

$$\bar{g}(u) = \bar{g}(u + 2\delta) \quad \forall u \in \mathbb{R}^n.$$

If $\delta \neq 0$, then $\bar{g}$ is a nonzero periodic function, which contradicts the aperiodicity in condition (C2). Therefore $\delta = 0$, i.e., $\xi = \xi'$.

With $\xi = \xi'$, equation (3.39) reduces to $\Phi(x)^\top \theta = \Phi(x)^\top \theta'$ for all x, so $\Phi(x)^\top (\theta - \theta') = 0$ for all x. By the linear independence of the basis functions in condition (C3), this implies $\theta = \theta'$. □

3.8.3 Verification for the Symmetrized Kernel Model

We verify conditions (C1)–(C3) for the model as deployed. The Gaussian process specifies a distribution over functions, but the EM algorithm fits and compares specific functions: the per-candidate posterior means (3.20), each the prior mean μ_f plus a linear combination of the kernel functions $k(\cdot, v_j)$ centered at the trajectory points $v_j = x_0^{(i)} + s(t)$. Draws from the GP prior never enter the algorithm, and they are rougher objects that fall outside the reproducing kernel Hilbert space of k almost surely [99, Section 4]. The conditions therefore concern the posterior means, and Theorem 3.2 applies once every estimate is even (C1), no nonzero estimate is periodic (C2), and the coefficients of an estimate are unique (C3). We check these properties in turn, with k the symmetrized kernel (3.13) built from the squared exponential k_0 and with centers $v_1, \ldots, v_r$ collected over the candidates under comparison and distinct up to reflection.

(C1) Replacing x by $-x$ permutes the four terms of (3.13), so $k(-x, v) = k(x, v)$ and every kernel function is even. The prior mean is even by the symmetric fit, so every estimate is even.

(C2) Every estimate vanishes at infinity, since the kernel functions are finite sums of squared exponential functions and the fitted prior mean is zero outside the bounded grid of its piecewise linear bases. Any prior mean that vanishes at

infinity suffices here, so the piecewise linear choice is not essential. A periodic estimate g, with $g(x) = g(x + 2\tau\delta)$ for some $\delta \neq 0$, all $x \in \mathbb{R}^n$, and all $\tau \in \mathbb{N}$, must then be zero: the argument $x + 2\tau\delta$ leaves every bounded set as $\tau \to \infty$, so $g(x) = \lim_{\tau\to\infty} g(x + 2\tau\delta) = 0$ for every x.

(C3) Two estimates that coincide as functions have the same prior mean term, so their combinations of kernel functions also coincide, and uniqueness of the coefficients reduces to linear independence of the kernel functions. Squared exponential functions with distinct centers are linearly independent, since multiplying a linear combination that is identically zero by $\exp(\frac{1}{2}x^\top L^{-1}x)$ reduces the claim to the linear independence of the exponentials $x \mapsto \exp(x^\top L^{-1}v)$ with distinct v. Reflected centers give the same kernel function, $k(\cdot, -v) = k(\cdot, v)$, which is why the centers are taken distinct up to reflection: one of any reflected pair can be dropped without changing the estimate.

These three properties are exactly what the proof of Theorem 3.2 uses, so the joint estimation problem underlying the EM algorithm is globally identifiable.

II

Image-Based Calibration and Control of TEM

4

Tilt-Based Aberration Estimation in Transmission Electron Microscopy

Abstract - Transmission electron microscopes (TEMs) enable atomic-scale imaging but suffer from aberrations caused by lens imperfections and environmental conditions, reducing image quality. These aberrations can be compensated by adjusting electromagnetic lenses, but this requires accurate estimates of the aberration coefficients, which can drift over time. This chapter introduces a method for the estimation of aberrations in TEM by leveraging the relationship between an induced tilt of the electron beam and the resulting image shift. The method uses a Kalman filter (KF) to estimate the aberration coefficients from a sequence of image shifts, while accounting for the drift of the aberrations over time. The applied tilt sequence is optimized by minimizing the trace of the predicted error covariance in the KF, which corresponds to the A-optimality criterion in experimental design. We show that this optimization can be performed offline, as the cost criterion is independent of the actual measurements. The resulting non-convex optimization problem is solved using a gradient-based, receding-horizon approach with multi-starts. Additionally, we develop an approach to estimate specimen-dependent noise properties using expectation maximization (EM), which are then used to tailor the tilt pattern optimization to the specific specimen being imaged. The proposed method is validated on a real TEM setup with several optimized tilt patterns. The results show that

This chapter is based on van Hulst et al. [103].

optimized patterns significantly outperform naive approaches and that the aberration and drift model accurately captures the underlying physical phenomena. A direct comparison with the widely used Zemlin tableau shows that the proposed method achieves comparable or higher image quality on amorphous specimens, while additionally extending to non-amorphous specimens where the Zemlin tableau cannot operate.

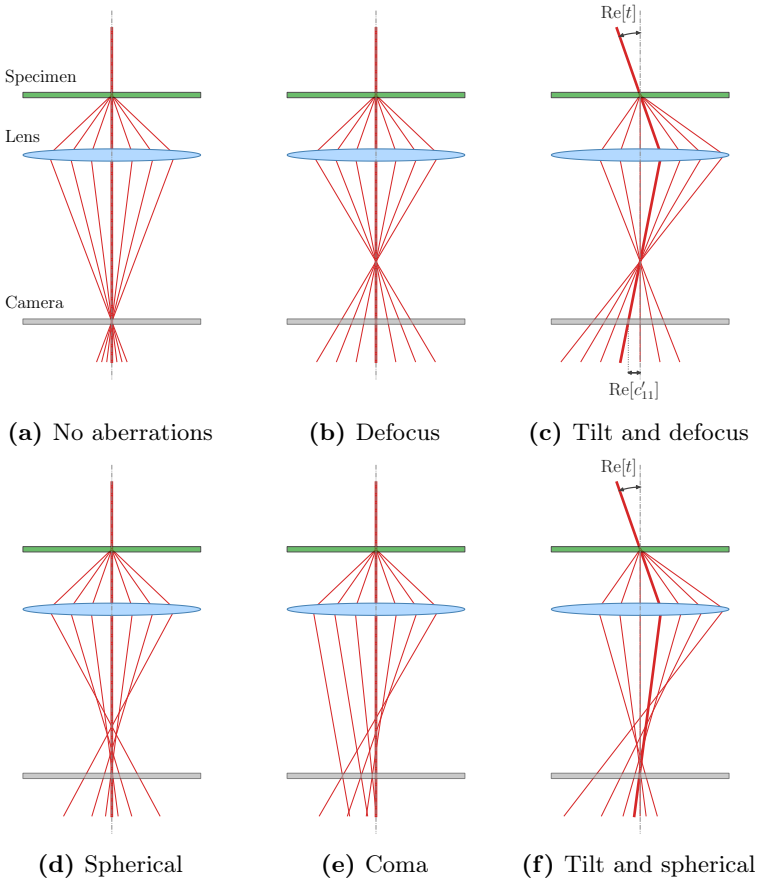
4.1 Introduction

Transmission electron microscopy (TEM) has enabled and continues to enable scientific progress across numerous fields, including materials science, semiconductors, and life sciences. A notable example of its impact is in the development of the COVID-19 vaccine, where TEM played a critical role by enabling the visualization of the viral spike protein [104]. What makes the TEM so powerful is its ability to produce images at the atomic scale. This capability stems from the use of electrons, which have significantly shorter wavelengths than visible light. Since resolution is inversely proportional to wavelength, the shorter the wavelength, the higher the attainable resolution.

TEM images are created by directing a beam of electrons onto a specimen and recording the electrons after they have interacted with it. However, accurately guiding the electrons is a non-trivial task. The electron beam is shaped and focused by electromagnetic lenses, which inherently suffer from high sensitivity to environmental conditions and manufacturing inconsistencies. The resulting imperfections in the electron beam are known as aberrations, see Figure 4.1 for a schematic representation of several aberration types. Under the presence of aberrations, the resulting TEM images suffer from reduced contrast, loss of resolution, and blurring.

Fortunately, the aberrations can be compensated by adjusting the electron optical components such as electromagnetic lenses, deflectors, and stigmators. However, this compensation is not straightforward, as the aberrations cannot be measured directly. Additionally, the aberrations drift over time due to thermal effects, mechanical vibrations, and electromagnetic field fluctuations. This limits both image quality and TEM throughput. As a result, fast and accurate estimates of the aberration values are highly desirable.

A well-known principle from optics is that tilting the electron beam in the presence of aberrations causes an apparent image shift in the x, y-plane [105], revealing information about the underlying aberration coefficients, see Figure 4.1 for an illustration of the aberrations and the tilt-shift interaction. This principle has been employed in several aberration estimation approaches. Classical methods such as the Zemlin tableau analyze diffractograms acquired at different beam tilts [15], while more direct approaches estimate aberrations from the tilt-induced image shifts themselves [54, 55]. However, these methods face significant limitations: diffractogram-based approaches are restricted to specimens that produce suitable diffractograms (typically requiring amorphous regions), limiting their applicability to only a subset of specimen types. Additionally, they are time-consuming. Existing image-shift methods are limited by lengthy image acquisition and processing times that restrict the number of tilts that can be applied. Crucially, none of these approaches model specimen drift explicitly [106], making it difficult to separate drift effects from aberration changes.



(a) No aberrations **(b)** Defocus **(c)** Tilt and defocus

(d) Spherical **(e)** Coma **(f)** Tilt and spherical

Figure 4.1. Schematic illustration of beam tilt and aberrations in TEM. In each panel, an electron beam (—) hits the specimen (■) at a point on the optical axis (- - -). The scattered electrons are represented by seven rays. These rays propagate through an electromagnetic lens (●) and arrive at the camera (■). In the aberration-free case **(a)**, all rays converge to a single point. Under defocus **(b)**, the rays converge before the camera plane, and combining that defocus with beam tilt **(c)** produces an image shift $\mathrm{Re}[c'_{11}]$. Spherical aberration **(d)** over-deflects the outer rays, while coma **(e)** deflects rays asymmetrically. Combining spherical aberration with beam tilt **(f)** produces a coma-like pattern. In **(c)** and **(f)**, the arc above the specimen indicates the beam tilt angle $\mathrm{Re}[t]$ (formally defined in Section 4.2). Note that beam tilt, image shift, and coma each have an additional component in the orthogonal direction, not shown here. The ray trajectories follow from a thin-lens model; component distances are simplified for clarity.

This chapter introduces a method for tilt-based aberration estimation that addresses these limitations. Fast GPU-based image shift tracking [56] enables rapid acquisition of many tilt-induced shifts, which are processed in a Kalman filtering (KF) framework that naturally fits the physical model and accounts for aberration drifting. Importantly, the speed and accuracy of the resulting estimates depend on the choice of beam tilt sequence, which motivates the core question addressed in this research: *what sequence of tilts results in the fastest and most accurate aberration estimates?* The problem of designing measurement sequences to optimize estimation performance has been studied extensively in sensor scheduling and experiment design [107, 108], including adaptations to Kalman filtering for linear discrete-time systems [109], continuous scheduling parameters [110, 111], and receding-horizon approaches [112]. However, these works are not tailored to TEM aberration estimation, which involves the freedom to select a continuous, two-dimensional scheduling parameter at every time step and a non-convex but differentiable measurement model.

We address these challenges by formulating the tilt sequence selection as a sensor scheduling problem and proposing a tailor-made solution. Aberrations are estimated using standard Kalman filtering, and the tilt sequence is optimized by minimizing the trace of the predicted covariance. The optimization uses gradients from multiple starting points with receding-horizon optimization. The main contributions are:

- A formulation of the tilt-based aberration estimation problem as a sensor scheduling problem, extended to include specimen drift.
- Efficient, receding-horizon gradient-based sensor scheduling optimization based on the trace of the predicted covariance matrix in the Kalman filter.
- Estimation of specimen-dependent properties using expectation maximization (EM) to tailor the optimization and estimation.
- Experimental validation of optimized patterns on a real TEM setup, including a quantitative comparison with state-of-the-art Zemlin tableau methods using standard image quality metrics.

The rest of this chapter is organized as follows. Section 4.2 formulates the aberration estimation problem. Section 4.3 details the methods including the tilt-induced aberration model and tilt pattern optimization. Section 4.4 presents results on a real TEM setup. Section 4.5 concludes the chapter.

4.2 Problem Formulation

TEM forms high-resolution images by transmitting electrons (accelerated to 100-300 keV) through a specimen, see Figure 4.2. The transmission imprints information

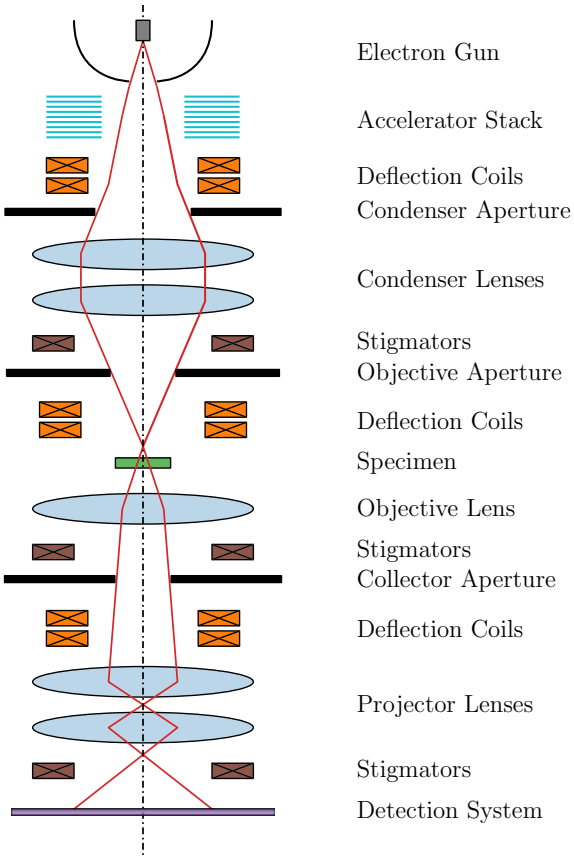


Figure 4.2. Schematic representation of TEM. An electron gun fires a beam of electrons into a stack of lenses and apertures. Between the lenses, there are stigmators and deflection coils which allow for active compensation of the lens aberrations. The transmitted electrons are collected after interaction with the specimen in a detection system. In contrast to STEM (Chapters 2 and 3), where the beam is focused to a probe and scanned across the specimen, the TEM beam is wide and illuminates the entire specimen simultaneously. The lenses here form a magnified image rather than a focused probe, and aberrations affect the phase of the electron wavefront and therefore image contrast and resolution. Note that the beam width is not drawn to scale; in reality it is on the order of a few Ångströms, while the distances between lenses and specimen are on the order of millimeters.

onto the electron wave by altering its phase. This imprint is enlarged by electromagnetic lenses, which project the electron wave onto a detector. The lenses are

inherently prone to imperfect focusing due to geometric and electromagnetic field characteristics and finite manufacturing precision. Additionally, environmental conditions such as temperature fluctuations and mechanical vibrations affect lens performance. These imperfections manifest as aberrations, altering the electron wavefront and reducing image resolution. Aberrations are characterized by the contrast transfer function (CTF), which describes how the electron wavefront evolves through the microscope [93, 113]. The CTF is defined as

$$\mathrm{CTF}(g) = E(g)\,\exp\left[-i\chi(g)\right], \tag{4.1}$$

where $g \in \mathbb{C}$ is the spatial frequency, $E(g)$ is the envelope function, and $\chi(g)$ is the wave aberration function. The variables are typically defined as complex numbers to represent 2D spatial coordinates. The envelope function $E(g)$ models coherence effects causing damping across frequencies and is assumed constant. For optimal contrast, $\exp\left[-i\chi(g)\right] \approx 1$ over a broad range of frequencies g. The wave aberration function $\chi(g)$ describes the phase shift of the electron wavefront and can be represented as

$$\chi(g) = \sum_{m,n,\ m-n=2p} \chi_{mn}(g), \tag{4.2}$$

where $m > 0$ and $p, n \geq 0$ are integers such that $m - n = 2p$. Each term χ_{mn} is a degree-m homogeneous polynomial in $\mathrm{Re}[g]$ and $\mathrm{Im}[g]$, and is linear in the aberration coefficient $c_{mn} \in \mathbb{C}$:

$$\chi_{mn}(g) = \frac{2\pi}{\lambda}\frac{\lambda^m}{m}\,\mathrm{Re}\Big[c_{mn}\cdot(g^*)^{(m+n)/2}\cdot g^{(m-n)/2}\Big], \tag{4.3}$$

where λ is the electron beam wavelength, g^* denotes the complex conjugate of g, and c_{mn} denotes the aberration coefficient with order m and foldness n (also called multiplicity). The influence of each term can be visualized using phase plates, see Figure 4.3. For a comprehensive overview of aberration types and naming conventions across different notations, we refer to Barthel [113, Chapter 2.2].

By adjusting the aberration coefficients c_{mn} appropriately, we can control the phase shift and thus the CTF. The coefficients can be adjusted by changing the settings of electron optical components including electromagnetic lenses. However, the coefficients are not directly measurable and must be estimated, which motivates the need for quick and accurate estimation methods.

4.2.1 Tilt-Induced Aberration Model

A key physical insight that can be used to estimate the aberrations is that deliberately tilting the electron beam interacts with the aberrations of the microscope.

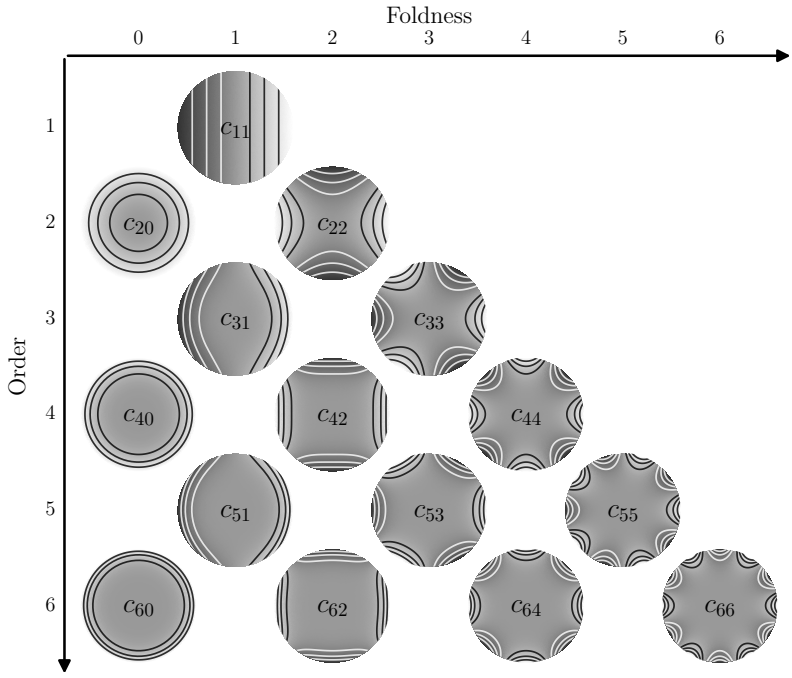


Figure 4.3. Phase plates of the single wave aberrations in TEM. The columns and rows correspond to the foldness and order of the aberration, respectively. Each phase plate displays the phase shift of the wavefront as a function of the spatial frequency $g \in \mathbb{C}$ in gray scale. Contour lines are plotted in white for positive phase shifts and in black for negative phase shifts. This figure has been adapted from Barthel [113, Chapter 2].

This interaction modifies the effective aberrations, which in turn leads to measurable changes in the image, such as an apparent shifting of the image [105]. We can express the effective aberrations as a function of the baseline aberrations and the beam tilt through

$$c'_{(\alpha+\gamma)(\alpha-\gamma)} = \rho\,(\alpha,\gamma) \times (\alpha+\gamma) \sum_{\beta=\alpha}^{M} \sum_{\delta=\gamma}^{\zeta(M,\beta)} \left[\binom{\beta}{\alpha} \binom{\delta}{\gamma} (t^*)^{\beta-\alpha}\, t^{\delta-\gamma} \right.$$
$$\left. \times \frac{c_{(\beta+\delta)(\beta-\delta)}}{\beta+\delta} + \binom{\beta}{\gamma} \binom{\delta}{\alpha} (t^*)^{\delta-\alpha}\, t^{\beta-\gamma} \frac{c^*_{(\beta+\delta)(\beta-\delta)}}{\beta+\delta} \right], \tag{4.4}$$

where $c_{mn} \in \mathbb{C}$ denotes the baseline aberration with order m and foldness n, $c'_{mn} \in \mathbb{C}$ denotes the effective aberration (following the tilt), and $t \in \mathbb{C}$ denotes the beam tilt, where t^* denotes the complex conjugate of t. M is the maximum order of baseline aberrations in the model. The indices $\alpha, \beta, \gamma, \delta$ are summation indices and the function $\zeta(M, \beta) = \min(\beta, M - \beta)$ limits the maximum foldness. The scaling factor $\rho(\alpha, \gamma)$ equals $\frac{1}{2}$ if $\alpha = \gamma$ and 1 otherwise. Note that the effective aberrations are linear in the baseline aberrations but generally nonlinear in the tilts. A concrete example for $M = 2$ is given in (4.6).

Among the effective aberrations in (4.4), the image shift c'_{11} is of particular interest because it can be directly estimated from TEM images. In practice, this effective shift c'_{11} is measured using a tracking algorithm based on cross-correlations between pairs of images, see e.g. van Horssen et al. [56], Bolme et al. [114]. Tracking works by constructing a so-called template image from one or more of the earlier acquired images. These approaches typically need a number of near-identical images to ensure the template is initialized correctly. In our use case this can be achieved in practice by starting with relatively small initial beam tilts, which ensures small image changes.

The tilt-induced image shift c'_{11} can be linearly related to the baseline aberration coefficients. To see this, we set $(\alpha + \gamma, \alpha - \gamma) = (1, 1)$ in (4.4) and evaluate the right-hand side for some maximum order M of baseline aberrations. We then collect the baseline aberration coefficients into a real vector $\mathbf{c} \in \mathbb{R}^{n_c}$ (where n_c is the number of real-valued aberration coefficients considered). Afterward, we can express the real and imaginary parts of c'_{11} as a linear function of $\mathbf{c}$:

$$\mathbf{c}' = \Psi(\mathbf{t})\mathbf{c} \tag{4.5}$$

with $\mathbf{c}' = \left(\mathrm{Re}\left[c'_{11}\right], \mathrm{Im}\left[c'_{11}\right]\right)^\top \in \mathbb{R}^2$, $\Psi(\mathbf{t}) \in \mathbb{R}^{2 \times n_c}$, and $\mathbf{t} := \left(\mathrm{Re}\left[t\right], \mathrm{Im}\left[t\right]\right)^\top \in \mathbb{R}^2$ the real-valued representation of the beam tilt t. For $M = 2$, we have $n_c = 5$ and obtain

$$\underbrace{\begin{pmatrix} \mathrm{Re}\left[c'_{11}\right] \\ \mathrm{Im}\left[c'_{11}\right] \end{pmatrix}}_{\mathbf{c}'} = \underbrace{\begin{pmatrix} 1 & 0 & \mathrm{Re}\left[t\right] & \mathrm{Re}\left[t\right] & \mathrm{Im}\left[t\right] \\ 0 & 1 & \mathrm{Im}\left[t\right] & -\mathrm{Im}\left[t\right] & \mathrm{Re}\left[t\right] \end{pmatrix}}_{\Psi(\mathbf{t})} \underbrace{\begin{pmatrix} \mathrm{Re}\left[c_{11}\right] \\ \mathrm{Im}\left[c_{11}\right] \\ c_{20} \\ \mathrm{Re}\left[c_{22}\right] \\ \mathrm{Im}\left[c_{22}\right] \end{pmatrix}}_{\mathbf{c}}. \tag{4.6}$$

From the equation, it is clear how beam tilts $\mathbf{t}$ reveal the underlying aberration coefficients $\mathbf{c}$ through measurable shifts $\mathbf{c}'$. Note that the estimation problem is complicated by the fact that the aberrations drift over time due to changing environmental conditions. Additionally, the particular tilt sequence has a significant impact on estimation speed and accuracy, motivating the search for optimal tilt sequences.

4.3 Methods

This section presents the methodology for aberration estimation and tilt sequence optimization in TEM. The tilt-induced shift equations with an aberration drifting model admit a linear Kalman filter (KF) formulation, which allows us to estimate the aberrations in a principled way. We then detail the tilt sequence optimization method, which is the main contribution, and develop an approach to estimate specimen-dependent measurement noise covariance using expectation maximization.

4.3.1 Dynamic Tilt-Aberration Model

The tilt-induced shift equations (4.6) can be used to predict the image shift in the presence of tilts and aberrations. However, the aberrations drift due to changing environmental conditions. For this reason, we introduce a dynamic model to improve estimation accuracy by relating aberrations to a sequence of measurements. This model includes two key physical effects.

First, the specimen may move with respect to the electron beam, a phenomenon known as specimen drift. This specimen drift is indistinguishable from changes in the image displacement aberration c_{11}. We model the drift by treating c_{11} as a time-varying state using a polynomial in time of order $b \in \mathbb{N}$. Since $c_{11} \in \mathbb{C}$ is a complex number representing drift in two dimensions, the polynomial requires b complex states (or equivalently $2b$ real states). Throughout the rest of this chapter, we use subscript $k \in \mathbb{N}$ to denote the discrete time index.

Second, the camera axis can in some cases be misaligned with respect to the tilt axis, resulting in relative rotation of tilt and shift coordinates. This is modeled by a state $\phi \in \mathbb{R}$ representing the rotational misalignment. We obtain a linear state-space form

$$x_{k+1} = Ax_k + \xi_k \tag{4.7}$$

$$y_k = C(\mathbf{t}_k)x_k + \varepsilon_k \tag{4.8}$$

with $x_k \in \mathbb{R}^d$ the state and $y_k = \mathbf{c}'_k \in \mathbb{R}^2$ the measurements, where $d = n_c + 1 + 2b$ and n_c is the number of aberrations considered. The states are normalized to typical values, as aberrations and drift terms differ in magnitude by several orders. This normalization ensures that the filter remains numerically well-conditioned and that the optimization criteria (used in Section 4.3.3) are not dominated by large-magnitude states. The process noise is $\xi_k \sim \mathcal{N}(0, \Sigma_\xi)$ and the measurement noise is $\varepsilon_k \sim \mathcal{N}(0, \Sigma_\varepsilon)$. For $b = 2$, which results in a second-order polynomial

model for specimen drift, we obtain

$$x_k = \left(\begin{array}{c} \mathbf{c}_k \\ \phi_k \\ \hline \mathrm{Re}\,[v_k] \\ \mathrm{Re}\,[a_k] \\ \mathrm{Im}\,[v_k] \\ \mathrm{Im}\,[a_k] \end{array}\right), \quad A = \left(\begin{array}{c|cccc} & \tau & \frac{1}{2}\tau^2 & 0 & 0 \\ & 0 & 0 & \tau & \frac{1}{2}\tau^2 \\ I_{n_c+1} & 0 & 0 & 0 & 0 \\ & & & \vdots & \\ & 0 & 0 & 0 & 0 \\ \hline & 1 & \tau & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ & 0 & 0 & 1 & \tau \\ & 0 & 0 & 0 & 1 \end{array}\right), \tag{4.9}$$

and

$$C(\mathbf{t}_k) = \left(\begin{array}{cc|cccc} \Psi\,(\mathbf{t}_k) & -\mathrm{Im}\,[t_k] & 0 & 0 & 0 & 0 \\ & \mathrm{Re}\,[t_k] & 0 & 0 & 0 & 0 \end{array}\right), \tag{4.10}$$

where $v_k \in \mathbb{C}$ and $a_k \in \mathbb{C}$ are the drift velocity and acceleration at time index k, with $\mathrm{Re}\,[v_k]$ and $\mathrm{Im}\,[v_k]$ representing the velocity components in the x and y directions, respectively, and similarly for $\mathrm{Re}\,[a_k]$ and $\mathrm{Im}\,[a_k]$. τ denotes the sampling time. Here, $\mathbf{c}_k$ is the time-varying counterpart of the aberration vector $\mathbf{c}$ introduced in (4.6), where the subscript k reflects that aberrations are now treated as a dynamic state. Essentially, the aberrations $\mathbf{c}_k$ (elements of x_k) remain constant (in the absence of process noise ξ_k), except for the shift aberrations $\mathrm{Re}\,[c_{11}]$ and $\mathrm{Im}\,[c_{11}]$, which are modeled as second-order polynomials in time. Note that the model can be easily extended to include polynomial drifting up to arbitrary orders of other aberrations, which would come at the cost of a more complex estimation problem. We assume the normalized initial state $x_0 \sim \mathcal{N}(0, I_d)$.

4.3.2 Kalman Filtering for Tilt-Based Aberration Estimation

To estimate the state x_k using the model (4.7), (4.8) based on a sequence of measurements $y_0, \ldots, y_k$, we employ the Kalman filter. The variable $\tilde{x}_{k|l}$ denotes the estimate of state x_k given measurements up to time l, i.e., $\tilde{x}_{k|l} := E[x_k \mid y_0, \mathbf{t}_0 \ldots, y_l, \mathbf{t}_l]$. Similarly, $P_{k|l}$ is the estimation error covariance, $E[(x_k - \tilde{x}_{k|l})(x_k - \tilde{x}_{k|l})^T \mid y_0, \mathbf{t}_0 \ldots, y_l, \mathbf{t}_l]$. The filter is initialized with $\tilde{x}_{0|-1} = 0$ and $P_{0|-1} = I_d$. The standard KF recursively computes estimates through update and prediction steps (see Kalman [115] for details).

The estimation could alternatively be approached using batch Linear Least Squares (LLS). For linear Gaussian systems, the KF and LLS yield equivalent estimates when the same prior knowledge is incorporated (i.e., in LLS through regularization and weighting). However, we opt for the KF framework because it naturally models aberration drift through state derivatives and process noise ξ_k,

and intuitively incorporates prior beliefs via $\tilde{x}_{0|-1}$, $P_{0|-1}$, Σ_ε, and Σ_ξ. While these aspects can also be addressed in LLS, the Bayesian recursive nature of the KF handles them more naturally.

A key result is that the estimation error covariance P evolves independently of actual measurements [108, 115], enabling offline optimization. We can therefore batch all measurements and express the N-step covariances in closed form:

$$P_{N-1|-1} = A^{(N-1)} P_{0|-1} (A^\top)^{(N-1)} + \sum_{i=0}^{N-2} A^i \Sigma_\xi (A^\top)^i, \tag{4.11}$$

$$P_{N-1|N-1} = P_{N-1|-1} - \bar{P}_\star^\top \bar{C}^\top(\mathcal{T}_N) S^{-1}(\mathcal{T}_N) \bar{C}(\mathcal{T}_N) \bar{P}_\star, \tag{4.12}$$

with the lifted observation matrix

$$\bar{C}(\mathcal{T}_N) = \begin{bmatrix} C(\mathbf{t}_0) & & & \\ & C(\mathbf{t}_1) & & \\ & & \ddots & \\ & & & C(\mathbf{t}_{N-1}) \end{bmatrix}, \tag{4.13}$$

the prior covariance $\bar{P} \in \mathbb{R}^{Nd \times Nd}$ of the state trajectory $\begin{bmatrix} x_0^\top & \cdots & x_{N-1}^\top \end{bmatrix}^\top$, which has blocks

$$\bar{P}_{ij} = \begin{cases} A^{i-j} P_{j|-1}, & i \geq j, \\ P_{i|-1} (A^\top)^{j-i}, & i < j, \end{cases} \tag{4.14}$$

its last block column $\bar{P}_\star := \bar{P} E$ with $E := \begin{bmatrix} 0 & \cdots & 0 & I_d \end{bmatrix}^\top$, and

$$S(\mathcal{T}_N) = \bar{C}(\mathcal{T}_N) \bar{P} \bar{C}^\top(\mathcal{T}_N) + I_N \otimes \Sigma_\varepsilon, \tag{4.15}$$

where $\otimes$ denotes the Kronecker product, and $\mathcal{T}_N := \{\mathbf{t}_i\}_{i=0}^{N-1}$ is the full tilt sequence.

We note that the batched form (4.12) is a standard equivalent reformulation of the recursive Kalman filter [116], expressing the N-step posterior covariance directly in terms of the prior $P_{0|-1}$ and the full tilt sequence $\mathcal{T}_N$. This batched KF formulation shares structural similarities with moving-horizon estimation (MHE) [117] and receding-horizon FIR filtering [118], which also process a batch of measurements. In our setting, however, we process the full measurement history from the start, whereas MHE and FIR typically use a sliding window of recent measurements. Additionally, in this work we use the batch form primarily for offline covariance prediction rather than real-time state estimation.

Given (4.11)–(4.15), we can compare tilt sequences in terms of their impact on estimation error covariance, which is a measure of uncertainty on the aberration values. The ability to predict the estimation error covariance enables the design of sequences that minimize this uncertainty.

4.3.3 Optimization of Tilt Sequences

Sensor scheduling involves actively deciding sensor configurations over time to optimize an objective, typically related to estimation performance. By choosing the beam tilt $\mathbf{t}_k$, we change the matrix $C(\mathbf{t}_k)$ in (4.8). The goal is to determine the N-step tilt sequence $\mathcal{T}_N$ that minimizes aberration estimation uncertainty using the predicted posterior covariance. We assess tilt pattern performance by taking a weighted trace of the predicted error covariance:

$$\begin{aligned} \min_{\mathcal{T}_N} \quad & \operatorname{tr}\left(W P_{N-1|N-1}\right), \\ \text{s.t.} \quad & (4.11),\ (4.12),\ (4.13),\ (4.14),\ (4.15), \\ & \|\mathbf{t}_k\|_2 \leq t_k^{\max}, \quad k \in \{0, 1, \ldots, N-1\}, \end{aligned} \tag{4.16}$$

where the tilt magnitude constraint will be discussed later. By taking the covariance trace as the cost, we essentially minimize the sum of variances of each of the aberration estimates (for $W = I_d$), known as A-optimality [108]. Other common optimality criteria in experiment design include D-optimality (minimizing the determinant) and E-optimality (minimizing the largest eigenvalue), which can be interpreted as minimizing the volume of the uncertainty ellipsoid and minimizing the worst-case variance, respectively. We choose A-optimality as it provides an appropriate overall uncertainty measure and is computationally convenient, though the results extend to other criteria. In batch experiment design using LLS, criteria are frequently derived from the Fisher Information Matrix (FIM), which is the inverse of the covariance matrix. In the KF framework, working directly with the posterior covariance is a natural approach.

The matrix $W \in \mathbb{R}^{d \times d}$ is a positive semi-definite weighting matrix ($W \succeq 0$) that can be used to prioritize estimation accuracy of certain aberrations over others. This weighting is applied *after* state normalization, ensuring users express preferences among comparable state values.

We constrain the tilt magnitude to be less than a user-defined scalar bound $t_k^{\max} \in \mathbb{R}_{>0}$, which varies with time index k. This allows us to address image tracker template initialization by limiting tilts at the beginning and increasing them as tracker accuracy improves. Additionally, the tilt magnitude must be upper-bounded since (4.4) is only valid for small tilts. Figure 4.4 shows an example of the tilt magnitude constraint as a function of time.

Several aspects make this sensor scheduling problem distinct from standard sensor scheduling problems:

1. The sensor scheduling parameter is a *continuous*, two-dimensional vector that can be selected at *every timestep*;
2. The cost function $\operatorname{tr}(W P_{N-1|N-1})$ is *differentiable* with respect to $\mathcal{T}_N$, enabling efficient gradient-based optimization;

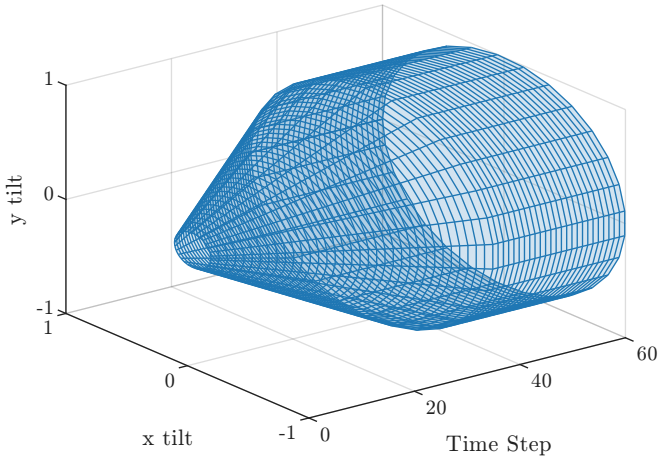


Figure 4.4. Example of a user-defined tilt constraint $t_k^{\max}$ as a function of time k. The red lines indicate the boundaries of the feasible set. Note that the tilt values are normalized.

3. Despite differentiability, the cost function is *non-convex* in the design variables, implying multiple local minima may exist;

4. The feasible set is convex and can be represented as a simple box constraint in polar coordinates.

We now detail how to solve the optimization problem (4.16) using these properties.

The first key insight is that we can take an analytical gradient with respect to the design variables, enabling efficient gradient-based algorithms. Using the identities

$$\frac{\partial(\operatorname{tr}(X(y)))}{\partial y} = \operatorname{tr}\left(\frac{\partial X(y)}{\partial y}\right), \tag{4.17}$$

and

$$\frac{\partial(X^{-1}(y))}{\partial y} = -X^{-1}(y)\frac{\partial X(y)}{\partial y}X^{-1}(y), \tag{4.18}$$

which hold for any vector y and invertible matrix-valued function $X(y)$, we obtain

$$\begin{aligned}\frac{\partial P_{N-1|N-1}}{\partial \mathcal{T}_N} = &-\frac{\partial G^\top(\mathcal{T}_N)}{\partial \mathcal{T}_N}S^{-1}(\mathcal{T}_N)G(\mathcal{T}_N) - G^\top(\mathcal{T}_N)S^{-1}(\mathcal{T}_N)\frac{\partial G(\mathcal{T}_N)}{\partial \mathcal{T}_N} \\ &+ G^\top(\mathcal{T}_N)S^{-1}(\mathcal{T}_N)\frac{\partial S(\mathcal{T}_N)}{\partial \mathcal{T}_N}S^{-1}(\mathcal{T}_N)G(\mathcal{T}_N),\end{aligned}$$

(4.19)

in which $G(\mathcal{T}_N) := \bar{C}(\mathcal{T}_N)\bar{P}_\star$ is the covariance between the measurements and x_{N-1}, with $\frac{\partial G(\mathcal{T}_N)}{\partial \mathcal{T}_N} = \frac{\partial \bar{C}(\mathcal{T}_N)}{\partial \mathcal{T}_N}\bar{P}_\star$, and

$$\frac{\partial S(\mathcal{T}_N)}{\partial \mathcal{T}_N} = \frac{\partial \bar{C}(\mathcal{T}_N)}{\partial \mathcal{T}_N}\bar{P}\bar{C}^\top(\mathcal{T}_N) + \bar{C}(\mathcal{T}_N)\bar{P}\frac{\partial \bar{C}^\top(\mathcal{T}_N)}{\partial \mathcal{T}_N}, \tag{4.20}$$

and

$$\frac{\partial \bar{C}(\mathcal{T}_N)}{\partial \mathcal{T}_N} = \begin{bmatrix} \frac{\partial C(\mathbf{t}_0)}{\partial \mathcal{T}_N} & & & \\ & \frac{\partial C(\mathbf{t}_1)}{\partial \mathcal{T}_N} & & \\ & & \ddots & \\ & & & \frac{\partial C(\mathbf{t}_{N-1})}{\partial \mathcal{T}_N} \end{bmatrix}. \tag{4.21}$$

The availability of an analytical gradient is preferred over numerical approximations for efficiency and accuracy. This analytical gradient can be used within a standard constrained optimization algorithm (e.g., a projected gradient method or an interior-point method) that explicitly handles the constraints. To do so, we transform each tilt $\mathbf{t}_k$ into polar coordinates (r_k, ψ_k), where r_k is the magnitude and ψ_k is the angle. The magnitude constraints $\|\mathbf{t}_k\|_2 \leq t_k^{\max}$ then become simple box constraints $0 \leq r_k \leq t_k^{\max}$, $0 \leq \psi_k < 2\pi$, which are readily incorporated into gradient-based solvers.

Next, since the number of design variables scales with the tilt pattern length (i.e., $2N$ scalar variables), we use receding-horizon optimization (RHO) with a horizon $H < N$ to solve a sequence of reduced-size problems. At each current time k, we solve:

$$\begin{aligned} \min_{\{\mathbf{t}(i)\}_{i=k}^{k+H-1}} \quad & \operatorname{tr}\left(W P_{k+H-1|k+H-1}\right), \\ \text{s.t.} \quad & \|\mathbf{t}(i)\|_2 \leq t_i^{\max}, \quad i \in \{k, k+1, \ldots, k+H-1\}, \\ & P_{k|k-1} \text{ given}, \end{aligned} \tag{4.22}$$

here, $\{\mathbf{t}(i)\}_{i=k}^{k+H-1}$ denotes the design variables in the reduced-horizon tilt sequence. After solving and obtaining the optimized solution $\{\mathbf{t}^\star(i)\}_{i=k}^{k+H-1}$, we store the first element $\mathbf{t}_k = \mathbf{t}^\star(k)$, calculate $P_{k+1|k}$, and repeat with $k \leftarrow k+1$ until $k = N$ (set $H = \min(H, N-k)$). This is illustrated in Figure 4.5. RHO, even with $H = 1$, can provide near-optimal performance [119] while reducing design variables from $2N$ to $2H$.

Since the cost function is non-convex, we employ a *multi-start strategy*, solving the RHO problem from many randomized initial feasible sequences and selecting the best solution. Starting from $k = 1$, we introduce *warm starts* by using the

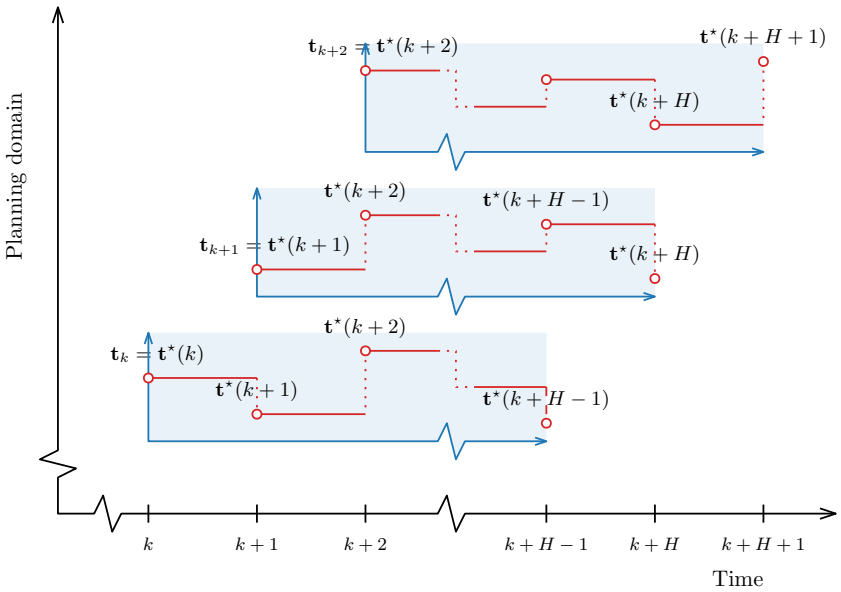


Figure 4.5. Schematic representation of receding-horizon optimization. Starting at time k, we solve an optimization problem with a reduced horizon H to find the optimal tilts $\{\mathbf{t}^\star(i)\}_{i=k}^{k+H-1}$. We then store the first tilt $\mathbf{t}_k = \mathbf{t}^\star(k)$, move up one time step in the planning domain and solve the $k+1$-th optimization problem with the updated predicted covariance $P_{k+1|k}$. We can use the previous solution $\{\mathbf{t}^\star(i)\}_{i=k}^{k+H-1}$ as a warm start for the next optimization problem. This process is repeated until we reach the end of the tilt sequence.

best previous RHO solutions as a subset of initial sequences. We use 1000 starting points total for all optimizations, with 100 warm starts when $k \geq 1$.

Note that the solution is sensitive to system parameters $P_{0|-1}$, Σ_ξ, and Σ_ε. While the first two can be chosen based on expert knowledge and historical data, the measurement noise covariance Σ_ε can vary significantly across specimens and imaging conditions. For this reason, we propose a method to estimate it which is detailed in the next section.

4.3.4 Estimating specimen-dependent measurement noise covariance

Different specimen types and imaging conditions such as field of view and defocus affect the amount of contrast and distinguishable features within the image. Since image tracker accuracy depends on these visual characteristics, different specimens

and conditions typically result in different measurement noise covariances Σ_ε.

To estimate the specimen-dependent Σ_ε, we employ expectation maximization, alternating between Kalman smoothing (E-step) and updating the noise covariance estimate (M-step). In the M-step, we update parameters by maximizing the expected complete-data log-likelihood [120]. For the measurement noise covariance Σ_ε, the update is given by

$$\Sigma_\varepsilon = \frac{1}{N}\sum_{k=0}^{N-1} \left(y_k - C(\mathbf{t}_k)\tilde{x}_{k|N-1}\right)\left(y_k - C(\mathbf{t}_k)\tilde{x}_{k|N-1}\right)^\top + C(\mathbf{t}_k)P_{k|N-1}C(\mathbf{t}_k)^\top, \tag{4.23}$$

where $\tilde{x}_{k|N-1}$ and $P_{k|N-1}$ are the smoothed state estimate and covariance, respectively, obtained from the Kalman smoother [120]. The EM algorithm is guaranteed to converge to a local maximum of the likelihood. It is advisable to run it multiple times with different initializations based on expert knowledge or rough estimates, and select the solution with the highest likelihood. This leads to the following adaptive filtering procedure.

Procedure 1. *Tilt-based aberration estimation and correction with specimen-dependent tilt pattern optimization*

1. Perform several tilt experiments with a general tilt pattern on the TEM and collect shift estimates.
2. Run the EM algorithm, alternating E-step and M-step (4.23), until convergence to estimate Σ_ε.
3. Determine weighting matrix W, tilt pattern length N, and horizon length H, then generate an optimized tilt pattern using the method in Section 4.3.3.
4. Apply the optimized tilt pattern to the TEM and gather shift measurements.
5. Use the Kalman filter to estimate the aberrations.
6. Compensate the aberrations by adjusting the electromagnetic fields.
7. Repeat Steps 4-6 as necessary.

4.4 Results

We present tilt pattern optimization results and experimental validation on a Thermo Fisher Scientific Krios TEM in Eindhoven, the Netherlands, following Procedure 1. A polycrystalline gold foil specimen is used for optimization and

validation, as it can handle high electron doses, allowing repeated application of different tilt patterns without damage.

We consider aberrations up to order $M = 4$ (14 coefficients). The specimen drift is modeled using a second-order polynomial ($b = 2$) for both x and y directions, plus image rotation, bringing the total number of states to $d = 19$.

4.4.1 Optimized Tilt Patterns

We apply the optimization method from Section 4.3 to find optimal tilt patterns for TEM aberration estimation. The measurement noise covariance Σ_ε for the gold foil is estimated using the EM algorithm described in the previous section. We use the interior-point algorithm to find local minimizers of (4.22) with tilt sequence length $N = 60$. Four tilt patterns are compared in Figure 4.6:

1. A Lissajous curve with a frequency ratio of 3:2;
2. A randomized pattern with tilts sampled uniformly from the time-dependent feasible set (see Figure 4.4);
3. An optimized pattern generated using gradient-based multi-start optimization with RHO, a horizon of $H = 1$ (greedy or myopic), and a weighting matrix $W = \frac{1}{d} I_d$;
4. An optimized pattern generated using gradient-based multi-start optimization with RHO, a horizon of $H = 10$, and a weighting matrix $W = \frac{1}{d} I_d$.

Patterns 1–2 are not optimized and are therefore expected to have poor performance in terms of the cost criterion. Pattern 3 (greedy optimization, $H = 1$) typically trades some performance for computational speed. Pattern 4 is optimized with $H = 10$ and is expected to perform well across most aberration types. The computation times for different optimization configurations are summarized in Table 4.1.

Figure 4.7 shows the trace of the weighted estimation error covariance $WP_{k|k}$ as a function of time index k. The 10-step optimized pattern 4) is expected to have the best performance in terms of the equally-weighted cost criterion. Interestingly, the greedy pattern 3) outperforms all other patterns at the beginning and middle of the estimation process, but falls behind pattern 4) slightly toward the end. This behavior is consistent with the fact that RHO is suboptimal by design: only the first tilt of each optimized horizon is applied, which favors immediately informative tilts. Note that for the greedy pattern, all intermediate patterns of lengths 1 to N are available without re-optimization, which is a practical advantage. While all patterns significantly reduce the estimates' covariance over time, the optimized patterns typically achieve the same performance in a much shorter time span. For

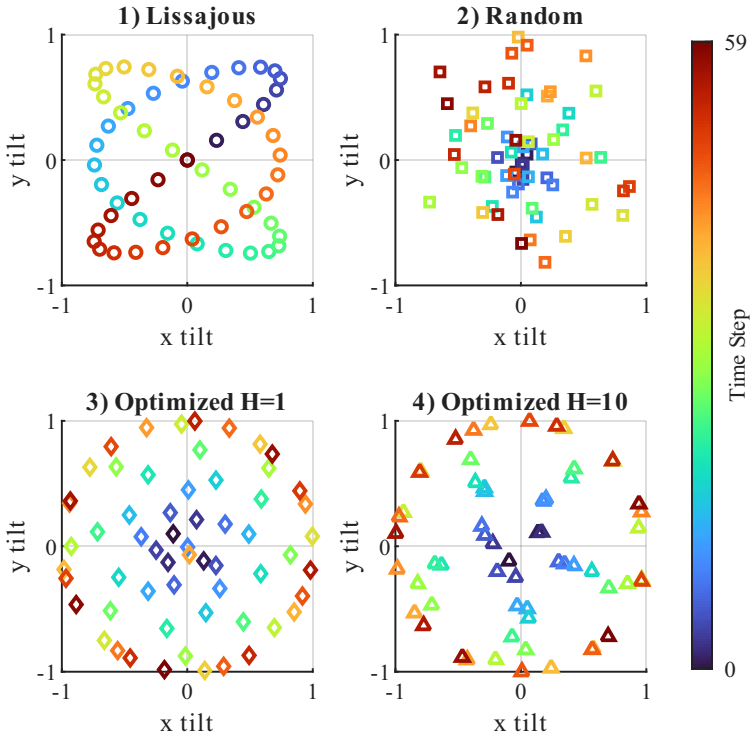


Figure 4.6. Comparison of different normalized tilt patterns. The patterns are: 1) Lissajous curve with frequency ratio 3:2, 2) completely randomized feasible pattern, 3) optimized pattern with $H = 1$, and 4) optimized pattern with $H = 10$. The color of each marker indicates the time step k, progressing from dark blue ($k = 0$) to dark red ($k = 59$).

example, the final covariance trace of the random pattern 2) is already reached after approximately 35 time steps with the optimized pattern 3), which represents a reduction of 25 time steps or 42%.

4.4.2 Experimental Validation

To assess the practical performance of the estimation method and the tilt patterns, we perform over 2000 trials on an actual TEM setup at different specimen locations. At each location, we test the tilt patterns from Figure 4.6 in randomized order to avoid bias from time-dependent effects.

To validate aberration estimation performance, we compare each experiment's estimated aberrations to the mean aberration estimates from experiments with

Table 4.1. Approximate wall-clock times for the optimization of a 60-tilt sequence with different horizon lengths. Times measured in MATLAB on a laptop computer; actual times may vary with hardware and system load. Note that the optimization is performed offline and does not affect the real-time estimation process. [1]These settings correspond to pattern 3) in Figure 4.6. [2]These settings correspond to pattern 4) in Figure 4.6.

Horizon length H	**Number of Multi-starts**	**Comp. Time (seconds)**	**Cov. Trace** $\mathrm{tr}(WP_{N-1\|N-1})$
1	10	37.1	1.06e-2
1	100	117.6	9.60e-3
1[1]	1000	485.6	9.33e-3
2	100	140.3	9.32e-3
5	1000	2650.8	9.63e-3
10	100	912.4	9.32e-3
10[2]	1000	7643.2	9.12e-3

similar timings and locations. This local mean can be viewed as a ground truth for the aberration values at that time and location. While this metric is sensitive to imperfect modeling, it demonstrates the variance in aberration estimates and the consistency of different tilt patterns.

We compare the experimentally realized variance in aberration estimates to the expected state estimate variance, given by the diagonal elements of $P_{N-1|N-1}$ in the KF, in Figure 4.8. Several conclusions can be drawn. First, the expected state estimate variance from the KF quite closely resembles the realized variance in aberration estimates, signaling that the tilt-induced shift model is relatively accurate. Typically, realized variances are slightly higher than predicted, likely due to undermodeling of aberrations beyond $M = 4$ and non-polynomial drift. The defocus aberration (c_{20}) is notably different in this aspect, likely because defocus is retuned at different points during the experiments, violating the constant-aberration assumption. Second, optimized patterns generally outperform non-optimized patterns in terms of realized estimation variance across all aberration types. Third, the greedy pattern ($H = 1$) achieves comparable performance to the $H = 10$ pattern. In practice, the greedy pattern is preferred due to its computational efficiency and the availability of intermediate-length patterns without re-optimization.

Having validated the estimation performance, we next investigate whether the improved aberration estimates translate into measurable improvements in image quality.

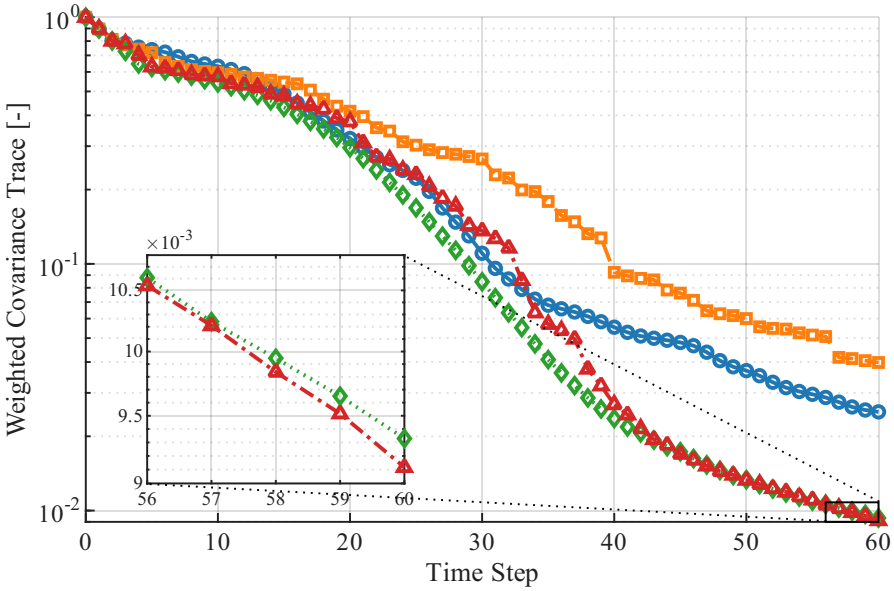


Figure 4.7. Weighted posterior covariance trace ($\mathrm{tr}(WP_{k|k})$) at each time index for selection of tilt patterns. The patterns are those of Figure 4.6: 1) Lissajous (—○—), 2) randomized (-□-), 3) optimized with $H = 1$ (·◇·), and 4) optimized with $H = 10$ (-△-). In this figure, the estimation error of every normalized state x_k is weighted equally using weight matrix $W = \frac{1}{d} I_d$. The inset focuses on the two optimized patterns at the final time steps, where the greedy pattern 3 very slightly underperforms the 10-step optimized pattern 4.

4.4.3 Image Quality Comparison

To quantitatively evaluate the practical benefit of the proposed method, we compare it against a standard Zemlin tableau [15] implementation in a Monte Carlo experiment on two specimen types: an amorphous carbon film and a polycrystalline gold specimen. In each trial, the microscope is intentionally set to a realistic aberrated state by perturbing defocus c_{20}, twofold astigmatism c_{22}, and axial coma c_{31}, and a *before* image is captured. Then, one of the two aberration correction methods is applied for several iterations of estimation and correction of these three aberrations, after which an *after* image is captured. For the proposed method, we use the optimized tilt pattern with $H = 10$ (pattern 4 in Figure 4.6). To ensure a fair comparison, the correction applied by the first method is reset before applying the second, so that both methods start from approximately the same aberrated state. This procedure is repeated 100 times for both methods and both specimen types.

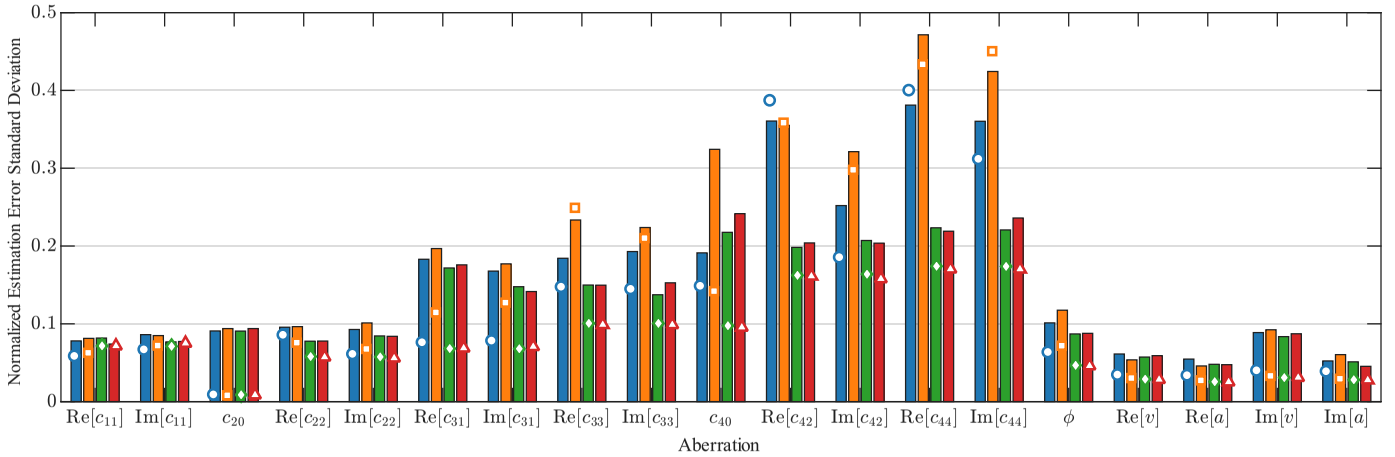


Figure 4.8. Comparison of the expected and realized standard deviation in aberration estimates for different tilt patterns applied to a gold specimen. The markers indicate the state estimation error predicted by the Kalman filter (i.e., the square root of the diagonal elements of $P_{N-1|N-1}$), while the bars indicate the realized standard deviation in aberration estimates across many experiments. Bars and markers are colored per tilt pattern, following Figure 4.6: 1) Lissajous (■, ○), 2) randomized (■, □), 3) optimized with $H = 1$ (■, ◇), and 4) optimized with $H = 10$ (■, △). Within each aberration, the four bars appear in this order from left to right.

Table 4.2. SNR values (cumulative signal power divided by cumulative noise power) for both methods and both specimen types (higher is better). This unitless metric summarizes overall image quality. The proposed method achieves higher SNR on both specimen types. The difference in values across specimen types is caused by the fact that amorphous carbon and polycrystalline gold have different contrast.

Specimen Type	Before	Zemlin	Proposed
Amorphous carbon	0.168	0.169	0.198
Polycrystalline gold	0.716	0.719	4.372

We assess image quality using the spectral signal-to-noise ratio (SSNR) [121], which quantifies the ratio of signal power to noise power as a function of spatial frequency. For each condition (before correction, after Zemlin correction, and after the proposed correction), the 100 images across trials form a single stack. The SSNR is then computed from this stack. The reproducible content across images constitutes the signal, while the image-to-image variability constitutes the noise. This noise includes contributions from residual aberrations (which vary between trials and which we aim to minimize) as well as from detector noise (which remains constant across methods). Radially averaging the resulting 2D spectra yields 1D SSNR curves as functions of the scalar spatial frequency g.

The resulting SSNR curves show at which spatial frequencies the imaging performance improves after correction. To summarize image quality in a single number, we integrate the signal power spectrum (the numerator of the SSNR [121]) and the noise power spectrum (its denominator) separately over all spatial frequencies, and take their ratio. This yields a single unitless signal-to-noise ratio (SNR) that captures the overall image quality. The SNR values are reported in Table 4.2.

Figure 4.9 shows the SSNR comparison for the amorphous carbon specimen. Both methods increase the SSNR relative to the uncorrected baseline across all spatial frequencies. The proposed method consistently achieves higher values than the Zemlin tableau, demonstrating that optimized tilt-based estimation yields accurate aberration correction even on amorphous specimens, the specimen type for which the Zemlin tableau is specifically designed. The observed improvement over the Zemlin tableau on this specimen type is modest and may partly reflect the particular Zemlin implementation tested. The key observation is that the proposed method achieves at least comparable aberration correction quality on amorphous specimens while also extending to non-amorphous specimens, as shown next.

Figure 4.10 shows the corresponding results for the polycrystalline gold specimen. Here, the advantage of the proposed method is substantially larger. This is expected: diffractogram-based methods such as the Zemlin tableau rely on Thon

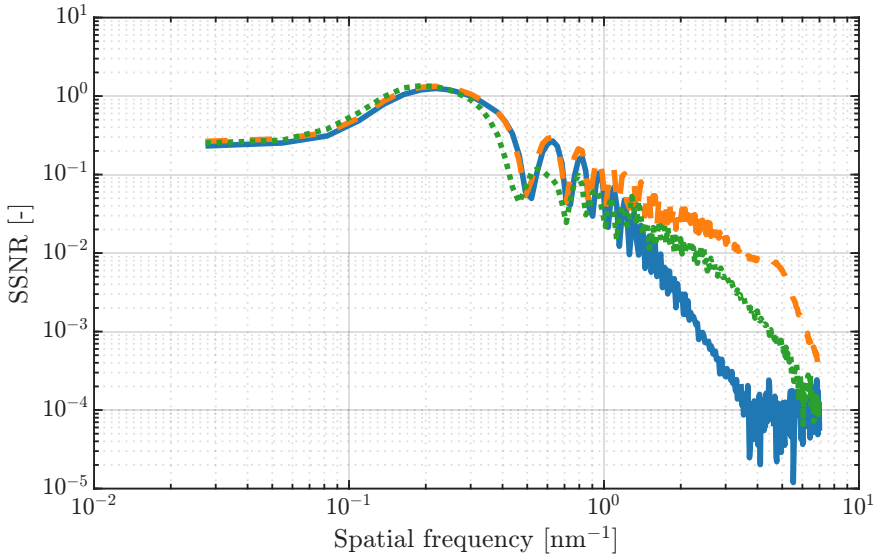


Figure 4.9. SSNR as a function of spatial frequency for the amorphous carbon specimen, computed from the 100 Monte Carlo trial images. Both the Zemlin tableau method (····) and the proposed method (- - -) improve the SSNR relative to the uncorrected baseline (——) at every frequency, with the proposed method achieving consistently higher values.

rings produced by amorphous regions, which are absent in polycrystalline specimens, and as a result the Zemlin tableau correction completely fails. The proposed method, relying solely on image shift measurements, is not subject to this limitation. This specimen-type independence is particularly relevant in life-science workflows, where aberrations are typically corrected on a polycrystalline gold substrate before moving the beam to the vitrified biological specimen of interest, minimizing electron dose on dose-sensitive material. Figure 4.11 illustrates this for an apoferritin specimen, a protein commonly studied in cryo-electron microscopy. After correcting aberrations on a gold region using the proposed method, the characteristic ring-like molecular structure of apoferritin becomes clearly visible, confirming that the aberrations have been sufficiently reduced to resolve the specimen's features.

4.5 Conclusion

Image-shift based aberration estimation in TEM has historically been limited by sensitivity to specimen drift and slow image acquisition and processing. This chapter

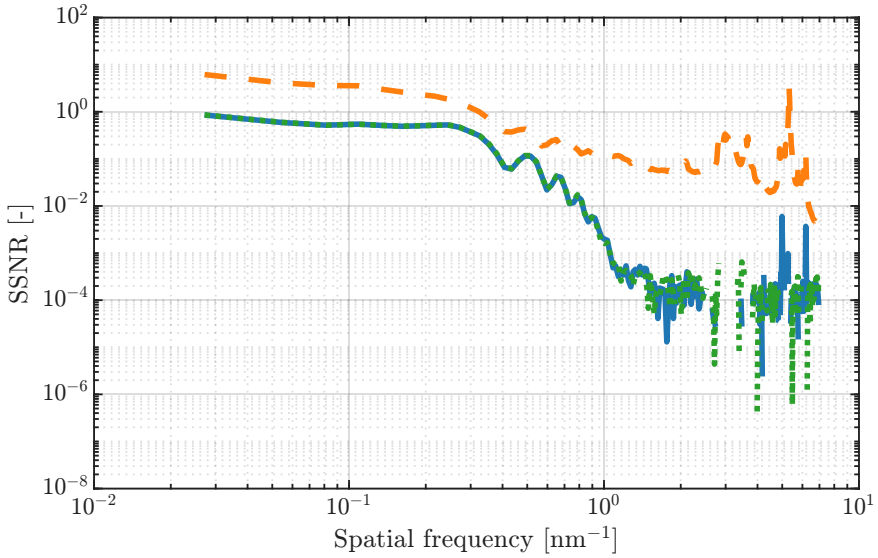
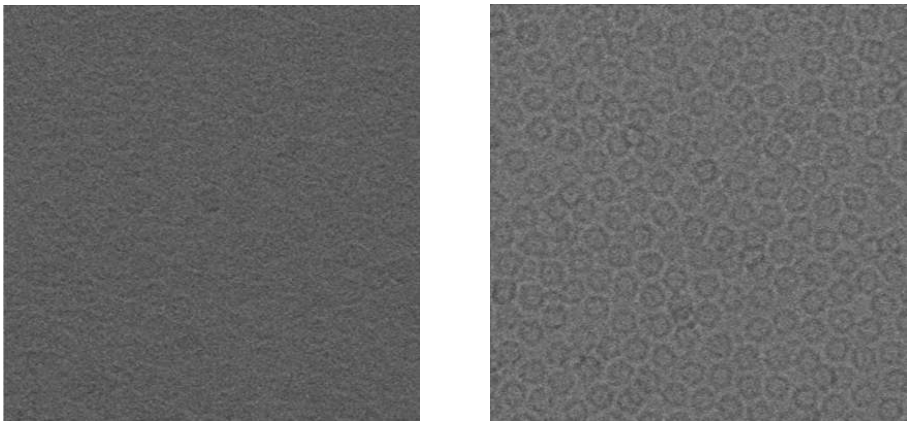


Figure 4.10. SSNR as a function of spatial frequency for the polycrystalline gold specimen, computed from the 100 Monte Carlo trial images. The proposed method (- - -) substantially improves the SSNR at every frequency on this specimen type, relative to the uncorrected baseline (——). The Zemlin tableau (····) fails on polycrystalline specimens and therefore does not improve the SSNR.

addresses both of these barriers. Fast GPU-based image shift tracking accelerates the measurement process, enabling the application of more tilts within a given time frame. The main contribution of this chapter is a method to optimize the sequence of applied tilts to maximize information gain. To this end, we formulate tilt-based aberration estimation as a sensor scheduling problem within a Kalman filtering framework that leverages the physical relationship between beam tilts and image shifts to estimate multiple aberration coefficients simultaneously. A key element of the framework is that it explicitly accounts for the temporal evolution of aberration coefficients, including specimen drift. The tilt sequence is optimized offline using an A-optimality criterion through gradient-based receding-horizon optimization with multiple starting points. We further propose an EM-based approach to estimate specimen-dependent measurement noise, tailoring the optimization to the specific specimen being imaged. The model predicts that optimized patterns achieve equal estimation accuracy in approximately 42% fewer time steps compared to naive patterns.

Experiments on a real TEM setup confirm the model predictions. The realized estimation variance across all tested tilt patterns closely matches the KF predictions,



(a) Before correction **(b)** After correction

Figure 4.11. Apoferritin specimen imaged before and after aberration correction using the proposed tilt-based method. In the uncorrected image, structural features are difficult to discern due to suboptimal contrast transfer. After correction, the characteristic ring-like molecular structure of apoferritin becomes clearly visible, indicating improved imaging conditions.

and optimized patterns substantially outperform naive ones. Beyond comparing tilt patterns, we also evaluate the overall method against the widely used Zemlin tableau. The proposed method achieves comparable or higher image quality on amorphous specimens. Importantly, the method also extends to non-amorphous specimens where the Zemlin tableau cannot operate, as demonstrated on a polycrystalline gold specimen.

A potential limitation is the reliance on the physical tilt-shift model, though the close match between predicted and realized estimation variance indicates limited model mismatch. The method also requires sufficient image features for reliable shift tracking, which may not hold for specimens with very uniform contrast. In such cases, the image tracker provides a natural diagnostic: anomalously small cross-correlation peaks between the template and the image can signal unreliable shift estimates that can be flagged or discarded. Beyond the TEM application, the sensor scheduling formulation and receding-horizon optimization approach may be of independent interest in other settings with continuous scheduling parameters and drifting system properties.

Future work could explore several promising directions. First, computational efficiency of the tilt sequence optimization could be improved by exploiting symmetries such as rotational invariance of the tilts or constraining the search space based on observed patterns in optimized tilt sequences. Second, extending the

optimization problem to include more accurate modeling of the image shift tracker could lead to the removal of the ad-hoc tilt magnitude constraints included to allow the tracker to initialize. Finally, data-driven sensor scheduling approaches could learn high-performing tilt policies directly from experimental data, potentially bypassing the need for exact analytical models.

5

Precision Specimen Positioning in Electron Microscopy through Hysteresis Compensation, Iterative Learning, and Vision-Based Sensing

Abstract - Electron microscopy requires nanometer-scale specimen positioning over a long stroke. Piezo-stepper actuators are well suited for this task, but their accuracy is limited by hysteresis, mechanical misalignments, and non-collocated sensing. Prior work has addressed these limitations on simplified lab setups. However, extending to a full electron microscope stage introduces coupled nonlinear kinematics and, importantly, the absence of a dedicated point-of-interest (POI) sensor. This chapter presents an integrated feedforward framework for precision positioning on such a stage inside an operational electron microscope. Per-element hysteresis compensation first linearizes the actuator response. In the absence of a dedicated POI sensor, a POI measurement is constructed from EM images through cross-correlation-based image tracking. From this measurement, we construct an encoder-based proxy for the POI position. Commutation-angle-domain iterative learning control then uses this proxy as its error signal to cancel the repeatable

This chapter is based on van Hulst et al. [122].

disturbances of stepping. Because the learned corrections are parameterized in the commutation angle, they transfer across the quasi-static range of drive frequencies. The framework reduces the POI tracking error by over 13× on the lab setup and by 7 to 12× on an operational transmission electron microscope.

5.1 Introduction

Electron microscopy (EM) requires nano-scale positioning of a specimen over a long stroke to enable high-resolution imaging [123]. An illustrative example of this is electron tomography, where the stage is tilted through a series of angles to reconstruct the three-dimensional structure of the specimen. The electron beam rarely passes exactly through the axis of rotation, so without correction the specimen drifts out of the field of view as the stage rotates. The stage must reposition the specimen at each tilt angle to hold it at the point of interest, and any residual error introduces artifacts into the reconstruction. The quality of the resulting images therefore hinges on positioning accuracy.

Piezo-stepper actuators (PSAs) are well suited for this task because they combine high stiffness with a large range of motion [124, 125]. Several parasitic effects, however, limit their positioning accuracy. Hysteresis in the piezoelectric material causes a nonlinear, history-dependent relationship between applied voltage and displacement. Mechanical misalignments arising from manufacturing tolerances introduce repeatable, cyclic errors during stepping. Non-collocated sensing, where encoders are located at the actuators rather than at the specimen, compounds these effects because quasi-static structural deformations cause discrepancies between the measured and true positions. Tilt-induced coupling between the axes further complicates positioning during tomographic acquisition.

Piezoelectric hysteresis has been studied extensively. Classical physics-based models such as those of Preisach, Prandtl–Ishlinskii, and Duhem provide analytical modeling frameworks [126, 127]. These models range from rate-independent formulations, which depend only on the input history, to rate-dependent ones that also account for the velocity of the applied voltage. Recent data-driven approaches tailored to piezo-stepper stages apply hysteresis compensation [128] and piezo waveform optimization [129]. Iterative Learning Control (ILC) has proven effective for canceling repeatable disturbances in precision positioning systems [130, 131]. The combination of hysteresis compensation and ILC is particularly powerful for piezo-stepper stages, where hysteresis and misalignment effects are often entangled [57].

Recently, van Meer et al. [57] demonstrated that combining rate-dependent hysteresis compensation with commutation-angle-domain ILC can reduce the RMSD of a single piezo-stepper to 8.7 nm at a drive frequency of 2 Hz. Although this represents a fifteenfold improvement compared to traditional feedforward, the study relied on a simplified 1DOF lab setup with a single actuator, trivial kinematics, and a direct capacitive sensor at the point of interest (POI), providing an idealized environment for feedforward design and validation.

Extending these results to the full EM stage presents several additional challenges. The stage has four degrees of freedom: three translational piezo-stepper

axes and one rotational axis. Nonlinear kinematics relate these to the 3D POI position. No direct position sensor is available at the POI inside the microscope. Structural deformation and tilt-induced coupling further cause the encoder-based estimate to differ from the true POI position. Image-based motion estimation is well established in electron microscopy, for example for correcting beam-induced specimen motion in cryogenic EM acquisition [132]. In particular, cross-correlation-based shift estimation can generate a POI position signal from EM images, as demonstrated in van Horssen et al. [56], van Hulst et al. [103]. In van Horssen et al. [56], this signal was further used for feedback at specimen level to counteract thermal drift. The central question addressed here is how to use such image-based observations for feedforward learning in an operational electron microscope, without a dedicated POI sensor.

This chapter presents an integrated framework for precision specimen positioning in an operational electron microscope, extending the methods of van Meer et al. [57] from a simplified single-PSA lab setup with a dedicated POI sensor to a real EM stage where no such sensor is available. The core new capability is to learn the feedforward corrections against a POI ground truth reconstructed from EM images, so that the same learning procedure runs with or without a dedicated sensor. The contributions are as follows.

- A commutation-angle-dependent sensor deviation is learned from the available POI measurement in each operating environment. On the lab setup, we use the capacitive reading p_c, and on the EM, we use the vision-based position estimate p_s. The deviation signal corrects encoder readings for quasi-static bending. The corrected estimate $\hat{p}$ provides a high-rate POI proxy that drives feedforward learning on both platforms.

- The commutation-angle-domain ILC of van Meer et al. [57] is extended from one PSA to three, with per-PSA error signals obtained from the corrected POI estimate $\hat{p}$ through the inverse kinematics. This transformation is identified from data and is non-trivial on the EM, where the measurement observes only the in-plane POI motion. We also reduce the hysteresis compensation to a two-parameter affine function of the history variable h, a simplification of the rate-dependent radial-basis-function model of van Meer et al. [57].

- All methods are validated on a full 4-degree-of-freedom lab setup and on an operational transmission electron microscope.

The remainder of the chapter is organized as follows. Section 5.2 describes the positioning stage, the coordinate systems, and the problem definition. Section 5.3 introduces the two sensing modalities and the sensor deviation. Section 5.4 presents the control design, covering hysteresis compensation and ILC. Section 5.5 shows experimental results and Section 5.6 concludes the chapter.

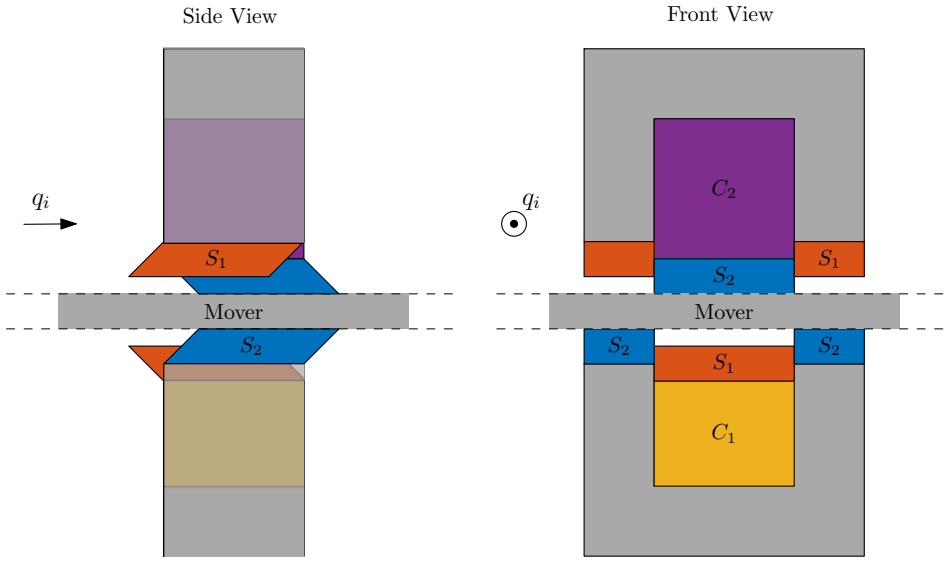


Figure 5.1. Schematic of a piezo-stepper actuator, following van Meer et al. [57]. The clamps (C_1, C_2) press the shear elements (S_1, S_2) onto the mover. When a shear element is in contact with the mover, it expands or contracts laterally to push or pull the mover in the q_i direction.

5.2 System Description and Problem Definition

The positioning stage is studied in two environments: inside the electron microscope, and on a lab setup that permits a direct measurement at the POI.

5.2.1 Positioning Stage

The full EM positioning stage consists of three piezo-stepper actuators ($i \in \{1, 2, 3\}$) and one rotational actuator providing a tilt angle $\varphi \in \mathbb{R}$ that rotates the full stage. Each piezo-stepper contains two pairs of clamp and shear piezoelectric elements (C_1, S_1, C_2, S_2). A stepping cycle is parameterized by a commutation angle $\alpha_i \in [0, 2\pi)$: in the first half-cycle, clamp C_2 presses shear S_2 onto the mover, which expands laterally to displace the mover. When S_2 reaches its end of stroke, C_1 engages S_1 and C_2 releases, creating a continuous walking motion with a net advance of approximately 3 µm to 4 µm per full cycle. Figures 5.1 and 5.2 illustrate the piezo-stepper architecture and the commutation-angle-dependent voltages applied to the elements.

Figure 5.3 illustrates the positioning stage in both operating environments. For

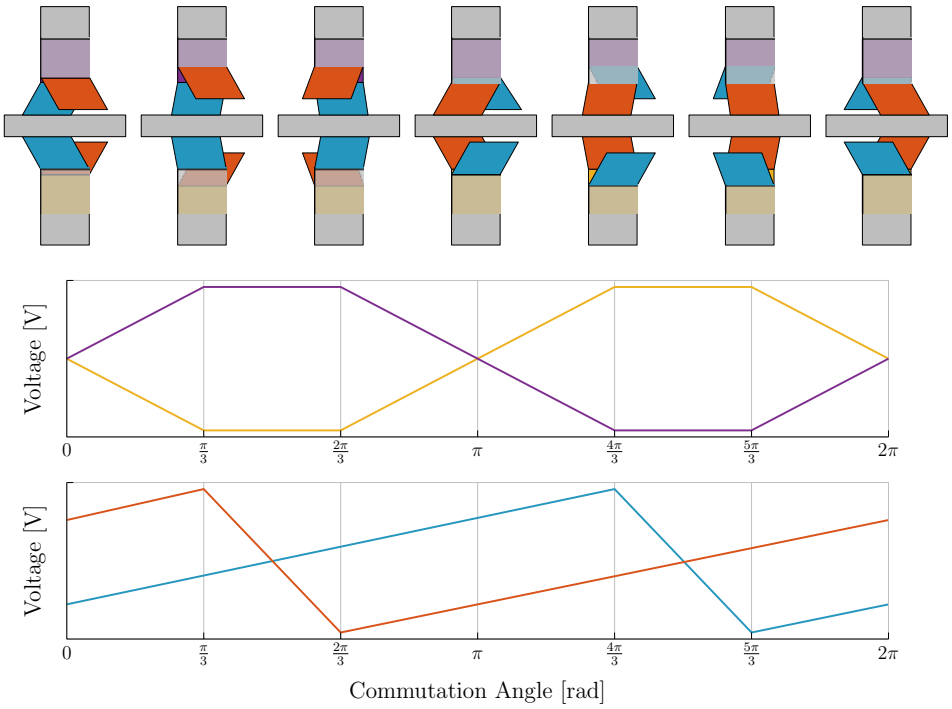


Figure 5.2. Voltage waveforms for piezo elements to achieve a stepping motion, following van Meer et al. [57]. The clamps (—,—) press the shears (—,—) onto the mover one by one, and the shears drag the mover along in the lateral direction.

model learning and validation, a separate 4DOF lab setup with the exact same geometry is used, which allows placement of capacitive sensors at the POI. These sensors provide a direct POI measurement, denoted $p_c \in \mathbb{R}^3$. No such sensor is available during EM operation. Instead, the microscope acquires images at a rate $F_v = F_s/N_v$, where F_s is the encoder sampling rate and N_v is the number of frames between vision updates. The value of N_v depends on the camera model and acquisition settings and is typically in the range of tens to several hundred. These images can be processed to obtain in-plane specimen shift estimates, as detailed in Section 5.3.1. This absence of a direct POI sensor inside the microscope is one of the central challenges in the present work.

Two position variables appear throughout this chapter. The encoder measurements $q = [q_1, q_2, q_3]^\top \in \mathbb{R}^3$ are linear displacements recorded at the three PSAs, which are imperfect observations of the true PSA positions subject to sensor noise. The POI coordinates $p = [p_x, p_y, p_z]^\top \in \mathbb{R}^3$ are the Cartesian position of the

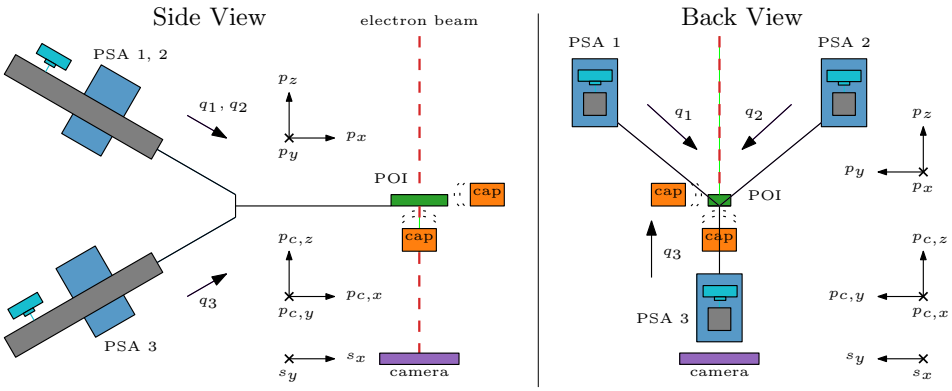


Figure 5.3. Schematic of the motion stage for both the lab setup and the electron microscope. The capacitive POI sensors (■) are present only in the lab setup and measure the POI through p_c, while the electron beam (- - -) and the EM camera (■) are present only in the microscope and measure the POI through p_s. The three piezo-stepper axes q_1, q_2, q_3 are shown, along with the POI position p.

specimen at the point of interest. The mapping between these two quantities is discussed in Section 5.2.2.

5.2.2 Coordinate Transformations

A nonlinear forward kinematics model relates the encoder readings to an estimated POI position:

$$\hat{p}_0 = \Phi(q, \varphi). \tag{5.1}$$

Here, Φ is a kinematic model, and φ is the tilt angle. The notation $\hat{p}_0$ emphasizes that this is a model-based estimate, not the true POI position p. Model errors, noise, and dynamic effects cause a discrepancy between $\hat{p}_0$ and p. The corresponding inverse kinematics model is denoted

$$\hat{q} = \Gamma(p, \varphi), \tag{5.2}$$

and maps a desired POI position to the piezo coordinates needed to reach it. The inverse kinematics is used later for projecting POI errors onto individual piezo coordinates (Section 5.4.2).

The kinematic maps Φ and Γ capture the nominal rigid-body geometry of the stage. The true POI position deviates from this prediction for several reasons. Contact forces during commutation elastically deform the mover, producing a quasi-static encoder–POI offset that is repeatable in the commutation angle α.

Structural flexibility, differential thermal expansion, and tilt-dependent coupling cause further deviations that are not α-periodic. The present work targets only the α-periodic offset, which is corrected by the sensor deviation of Section 5.3.3.

5.2.3 Problem Definition

The objective is to minimize the POI tracking error $e_p(k) = r_p(k) - p(k)$, where k is the discrete-time sample index, $r_p \in \mathbb{R}^3$ is the desired POI trajectory, and p is the true POI position. Since p is not directly measurable inside the microscope, a POI estimate p_e is used in its place, whose form depends on the operating environment and is detailed in Section 5.3. Because each PSA is actuated and learned individually, the error is assessed in piezo coordinates, where the reference and the estimate are mapped through the inverse kinematics and compared per axis,

$$\varepsilon_i(k) = [\Gamma(r_p(k), \varphi)]_i - [\Gamma(p_e(k), \varphi)]_i, \quad i \in \{1, 2, 3\}. \tag{5.3}$$

The kinematic map Φ is invertible on the stage workspace for each fixed φ, so Γ is a bijection and $\varepsilon_i = 0$ for all i exactly when $p_e = r_p$. Driving these per-axis errors to zero is therefore equivalent to nulling the POI error. Performance is then quantified per axis by the root-mean-square deviation

$$\sigma_i = \sqrt{\frac{1}{N} \sum_{k=1}^{N} \varepsilon_i(k)^2}. \tag{5.4}$$

5.3 POI Estimation

Iterative learning control requires a POI error signal at the encoder rate F_s. The encoders provide a signal at this rate, but contact forces during commutation elastically deform the mover and bias the reading relative to the true specimen position. Correcting this bias requires a POI ground-truth measurement against which the correction can be learned. On the lab setup this ground truth is the capacitive reading p_c. On the EM, where no such sensor exists, it is a vision-based estimate p_s derived from the EM images. Section 5.3.1 describes how p_s is obtained, Section 5.3.2 how the kinematic maps are calibrated, and Section 5.3.3 how the learned correction turns the encoder reading into a high-rate POI proxy $\hat{p}$.

5.3.1 Vision-Based Position Sensing

EM images provide an independent source of in-plane POI information. A cross-correlation-based tracker compares consecutive images and estimates the in-plane

specimen displacement [56, 103]. The tracker runs on GPU, enabling processing at the camera frame rate [103]. The output is a 2D inter-frame shift

$$s(k) = [s_x(k),\, s_y(k)]^\top \in \mathbb{R}^2, \tag{5.5}$$

where k indexes camera frames and $s_x(k)$, $s_y(k)$ are noisy measurements of the in-plane specimen displacement in the p_x and p_y directions since the previous frame. The tracker is sampled at $F_v = F_s/N_v$, substantially slower than the encoders.

Converting the pixel-level shift to physical length units depends on the optical settings of the microscope. Assuming parallel illumination is used, the dominant parameter is the selected magnification value. A large inter-frame shift broadens the correlation peak and can produce outliers. The specimen motions considered here are slow enough that the shift remains small and the tracker produces reliable estimates.

How well $s(k)$ covers the motion of each piezo axis depends on φ, because Φ couples the piezo axes to the in-plane and out-of-plane directions through the tilt. At $\varphi = 0$, all three axes contribute to the out-of-plane direction p_z, so $s(k)$ captures only a fraction of each axis's motion. At $\varphi \approx \pi/6$ rad, PSA 2's axis lies entirely in the imaging plane and its full motion is observed by $s(k)$. At $\varphi \approx -\pi/6$ rad, PSA 1's axis lies entirely in the imaging plane instead. During calibration, the tilt angle can be set to maximize the coverage of the axis under calibration. During tomographic acquisition, φ is prescribed by the imaging protocol and the coverage of each axis changes accordingly.

The specimen position relative to a reference frame k_0 is estimated by accumulating consecutive inter-frame shifts. Because the tracker observes only in-plane motion, the accumulation updates the two in-plane coordinates while the out-of-plane coordinate is held fixed:

$$p_s(k) = p_s(k_0) + \sum_{j=k_0+1}^{k} \begin{bmatrix} s(j) \\ 0 \end{bmatrix} \in \mathbb{R}^3. \tag{5.6}$$

The reference $p_s(k_0)$ is initialized from the encoder-based forward-kinematics estimate, $p_s(k_0) = \Phi(q(k_0), \varphi)$, which anchors p_s to an absolute POI position rather than to an arbitrary origin. Between camera frames, p_s is held at its most recent value. The resulting estimate $p_s \in \mathbb{R}^3$ serves as the POI ground truth for learning the sensor deviation, as described in the next section.

5.3.2 Kinematic Calibration

In practice, the kinematic maps are obtained by calibration. The experiments cover a small region of the workspace, over which Φ is well approximated by its linearization at the current tilt angle. Let $\bar{p}$ denote the available POI measurement,

with $\bar{p} = p_c$ on the lab setup and $\bar{p} = p_s$ on the EM. Since only the in-plane components of p_s are measured (5.6), the EM fit uses those two components. To identify the linearization, a calibration move displaces all three PSAs while $\bar{p}$ and q are recorded. Expressing both signals relative to the start of the move at sample k_0 removes the constant offset between the coordinate frames, and the linearized forward map follows from the least-squares fit

$$\hat{K} = \arg\min_{K} \sum_{k=1}^{N_K} \left\| \bar{p}(k) - \bar{p}(k_0) - K\big(q(k) - q(k_0)\big) \right\|_2^2, \tag{5.7}$$

where N_K is the number of recorded samples. The fit yields $\hat{K} \in \mathbb{R}^{3\times3}$ on the lab setup and $\hat{K} \in \mathbb{R}^{2\times3}$ on the EM.

On the lab setup, $\hat{K}$ is invertible, and $[\Gamma(\bar{p}, \varphi)]_i$ is evaluated by applying the i-th row of $\hat{K}^{-1}$ to $\bar{p} - \bar{p}(k_0)$ and adding $q_i(k_0)$. On the EM, $\hat{K}$ has no inverse: two measured components cannot separate three piezo coordinates. However, in all experiments only one PSA steps at a time, and its motion appears in the measurement along the corresponding column κ_i of $\hat{K}$. The piezo coordinate is therefore recovered by projecting onto this column, with the row vector $\kappa_i^\top / \|\kappa_i\|_2^2$ taking the place of the i-th row of $\hat{K}^{-1}$. This recovery is exact for motion of PSA i alone, and simultaneous stepping of several PSAs would require an out-of-plane measurement (Section 5.3.1).

5.3.3 Encoder-Based POI Estimation

Applying the forward kinematics (5.1) to the encoder readings gives the baseline estimate $\hat{p}_0 = \Phi(q, \varphi)$, available at the full encoder rate F_s. This estimate, however, does not account for a clamping-induced bending effect that arises during commutation. When a clamp engages the mover, the resulting contact force causes a small deformation of the mover, as illustrated in Figure 5.4. Because the encoder is not located at the end of the mover, it does not capture this deformation, and a discrepancy between the encoder reading and the true POI position arises. The effect is quasi-static: it depends on the current commutation angle but not on the stepping velocity, as confirmed by the data in Figure 5.5.

To correct for this discrepancy, a sensor deviation is applied in piezo coordinates before computing the forward kinematics. Because the deformation depends on the commutation angle, the deviation is parameterized as a function of α:

$$\hat{p} = \Phi\big(q + \Delta(\alpha),\, \varphi\big), \tag{5.8}$$

where $\Delta(\alpha) = [\Delta_1(\alpha_1),\, \Delta_2(\alpha_2),\, \Delta_3(\alpha_3)]^\top \in \mathbb{R}^3$ collects three scalar deviation functions, one per PSA. Each $\Delta_i : [0, 2\pi) \to \mathbb{R}$ is a piecewise-linear function stored in a lookup table that maps the commutation angle of PSA i to a scalar

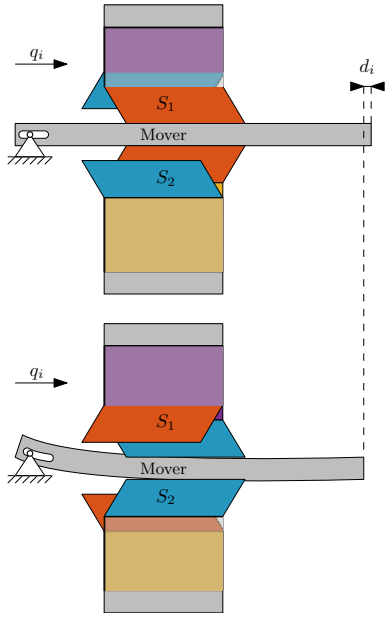


Figure 5.4. Illustration of the bending effect during clamp engagement. Contact forces between the clamp and shear elements cause local deformation that shortens the effective length of the mover. Because the encoder is located at the PSA, it does not capture this deformation, resulting in a commutation-angle-dependent discrepancy between the measured and true positions.

correction of its encoder reading. These scalar functions are obtained by projecting the POI estimation residual onto the individual piezo axes through the inverse kinematics Γ (5.2), as detailed below. The decomposition into independent per-PSA contributions is supported by the physical structure of the stage: the three movers are mechanically separate, so contact forces on PSA i do not propagate a bending deformation to the other two. Correcting q rather than p also keeps the correction scalar per PSA, whereas a correction in POI space would require a three-component vector.

The sensor deviation is learned from the POI measurement $\bar{p}$. For each PSA i, the discrepancy between $\bar{p}$ and the encoder reading is recorded over multiple commutation cycles while PSA i is stepping and the others are stationary. Although p_s is sampled more slowly than the encoder, the recording spans many commutation cycles, so the samples cover the full α grid. Transforming the POI residual to piezo axis i through the inverse kinematics Γ (5.2) gives the raw residual

$$d_i(k) = [\Gamma(\bar{p}(k),\, \varphi)]_i - q_i(k). \tag{5.9}$$

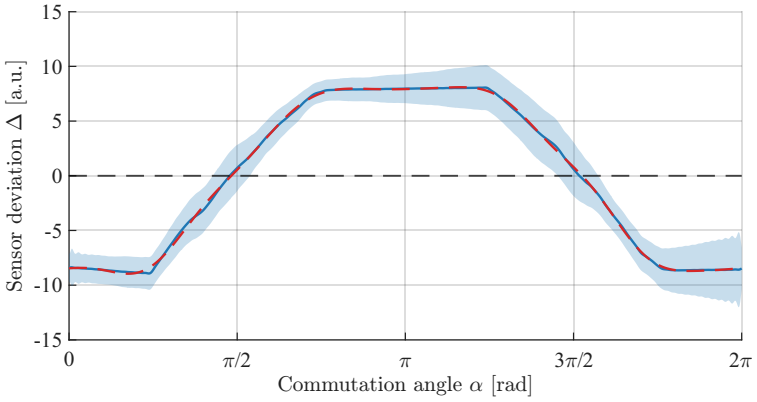


Figure 5.5. Mean projected encoder–POI residual d_i (5.9) (——) and learned piecewise-linear sensor deviation $\Delta_i(\alpha_i)$ (5.10) (- - -) versus commutation angle α_i for one PSA. The shaded area denotes $\pm\sigma$ of the residual samples.

The sensor deviation is parameterized as a linear combination of basis functions,

$$\Delta_i(\alpha_i) = \boldsymbol{\psi}_\Delta^\top(\alpha_i)\,\boldsymbol{\theta}_{\Delta,i}, \tag{5.10}$$

where $\boldsymbol{\psi}_\Delta(\alpha_i) \in \mathbb{R}^{n_\Delta}$ collects n_Δ basis functions and $\boldsymbol{\theta}_{\Delta,i} \in \mathbb{R}^{n_\Delta}$ are the coefficients. In practice, piecewise-linear basis functions on n_Δ equidistant grid points in $[0, 2\pi)$ are used for simplicity and interpretability. Stacking the basis function evaluations over all N_Δ recorded samples gives the matrix $\boldsymbol{\Psi}_\Delta = [\boldsymbol{\psi}_\Delta(\alpha_i(1)), \ldots, \boldsymbol{\psi}_\Delta(\alpha_i(N_\Delta))]^\top \in \mathbb{R}^{N_\Delta \times n_\Delta}$, and stacking the residual samples gives $\mathbf{d}_i \in \mathbb{R}^{N_\Delta}$. The coefficients are obtained by minimizing the squared residual:

$$\boldsymbol{\theta}_{\Delta,i} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^{n_\Delta}} \sum_{k=1}^{N_\Delta} \left(d_i(k) - \boldsymbol{\psi}_\Delta^\top(\alpha_i(k))\,\boldsymbol{\theta}\right)^2. \tag{5.11}$$

Because the model is linear in the parameters, the minimization reduces to a standard linear least squares problem with the closed-form solution

$$\boldsymbol{\theta}_{\Delta,i} = (\boldsymbol{\Psi}_\Delta^\top \boldsymbol{\Psi}_\Delta)^{-1} \boldsymbol{\Psi}_\Delta^\top \mathbf{d}_i. \tag{5.12}$$

Figure 5.5 shows the raw residual and the learned fit for a representative PSA on the lab setup.

Including the sensor deviation reduces the residual $\bar{p} - \hat{p}$ significantly compared to the baseline $\bar{p} - \hat{p}_0$, confirming that the quasi-static bending is the dominant source of discrepancy. The remaining residual is attributed to unmodeled effects such as flexible-mode vibrations, which are not captured by a static correction.

5.4 Feedforward Control Design

Figure 5.6 shows a block diagram of the control architecture. In the diagram, d_q denotes disturbances at the actuator level that are reflected in the encoder reading q, such as misalignment and handover transients, while d_p denotes disturbances between the actuator and the POI that the encoder does not capture, such as clamp-induced bending, thermal drift, and flexible-mode vibration. The control proposed in this work is entirely feedforward and acts within the commutation block, which generates the voltages applied to the piezoelectric elements. Two modifications are made there. Per-element hysteresis compensation (Section 5.4.1) inverts the voltage-to-displacement nonlinearity of each element. Commutation-angle-domain ILC (Section 5.4.2) then modifies the nominal commutation waveforms of Figure 5.2 to cancel the repeatable, α-periodic disturbances of stepping. Of the disturbances above, the feedforward cancels only the α-periodic parts, namely misalignment and handover transients through the ILC and clamp-induced bending through the sensor deviation. Both components are calibrated offline.

5.4.1 Per-Piezo Hysteresis Compensation

Hysteresis in the piezoelectric material causes a nonlinear, history-dependent relationship between applied voltage and element displacement. The waveform generator must invert this relationship to translate desired displacements into applied voltages accurately. This inversion is a prerequisite for the ILC of Section 5.4.2. The ILC corrects disturbances that are periodic in the commutation angle α, but uncompensated hysteresis introduces errors that depend on the voltage history rather than on α. These history-dependent errors are not α-periodic and therefore corrupt the ILC error signal [57].

The compensation is applied independently to each of the three PSAs, where each PSA contains multiple piezoelectric elements. Following van Meer et al. [57], we treat the clamp elements separately per movement direction and treat the shear elements direction-independently. Every element has the same architecture, so the model structure and identification procedure are identical for all elements. The equations below are written for a single generic element, so PSA and element indices are omitted for simplicity.

The voltage-displacement relation of an element traces a loop rather than a single curve. Such loops are captured by a range of models, from the operator-based Preisach [126] and Prandtl–Ishlinskii [127] models to algebraic descriptions such as the Ramberg–Osgood model [133]. For a piezo-stepper the loop is set by the most recent reversal, so the relevant history is the excursion

$$h(t) = |u(t) - u(\bar{t})| \tag{5.13}$$

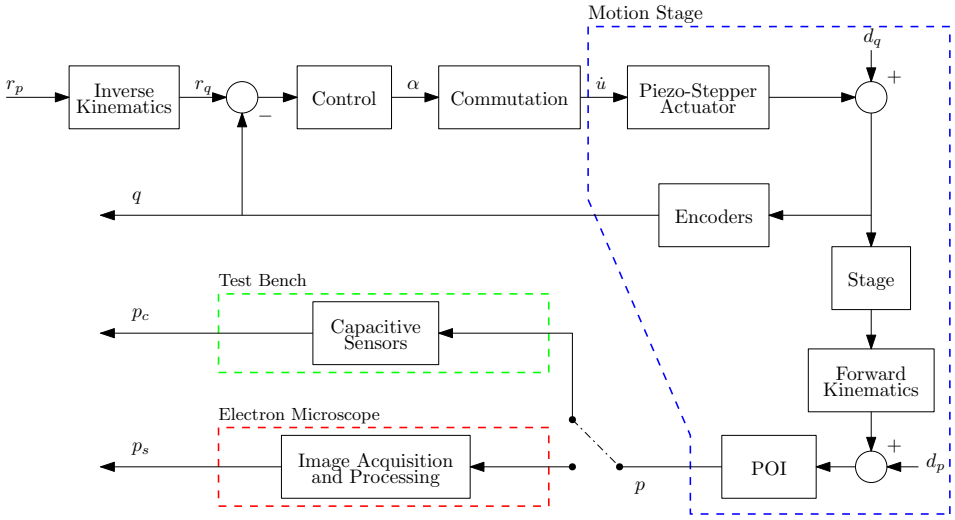


Figure 5.6. Block diagram showing the signal flow of the system. A POI reference r_p is fed into the inverse kinematics Γ to obtain the piezo reference r_q. This reference is tracked by low-level controllers. The PSAs obtain voltage signals from the commutation block, which can include hysteresis compensation and ILC-modified waveforms. The encoder signal q provides a measurement of the piezo positions, while the POI position p is either measured directly by the capacitive sensors p_c in the lab setup, or estimated from the EM images as p_s in the microscope.

of the voltage since that reversal at time $\bar{t}$. Following the memory-element model of Strijbosch et al. [128], the displacement rate is the voltage rate scaled by an incremental gain that depends on this history,

$$\dot{y} = M(h)\,\dot{u}, \tag{5.14}$$

with element displacement y, applied voltage u, and incremental gain $M > 0$. This is a special case of that memory-element model, which on each branch coincides with the Ramberg–Osgood hysteresis.

The gain cannot be measured directly, as the individual element displacements are not sensed. The element current stands in for the displacement, since the displacement rate is proportional to the current, $\dot{y} = \xi\,\iota$, with ξ an unknown constant [134]. Used in the incremental model (5.14), the ratio of current to voltage rate follows the gain up to ξ,

$$m := \left|\frac{\iota}{\dot{u}}\right| \approx \frac{M}{\xi}, \tag{5.15}$$

with per-sample values $m(k)$. The proxy uses only the element's own voltage and current, so each element is identified on its own and no position measurement is needed.

To identify the gain, each element is driven open-loop with steady-state voltage sweeps over drive frequencies from 0.1 Hz to 50 Hz, while its voltage and current are recorded. Figure 5.7 shows the resulting proxy m against the history h. The measurements fall along a single affine trend in h, with R^2 from 0.944 to 0.985 for the clamps and from 0.946 to 0.952 for the shears, so the rate dependence of the gain is small.

We adopt this affine gain,

$$\hat{M}(h) = \theta_1\, h + \theta_2, \tag{5.16}$$

with intercept $\theta_2 > 0$, the gain just after a reversal, and slope $\theta_1 \geq 0$, the rate at which it grows with the excursion h. Because the sweeps span a range of voltage rates and still share this single trend, a rate-independent gain suffices here. This affine gain simplifies the rate-dependent model of van Meer et al. [57] by dropping the dependence on the voltage rate.

In discrete time, the element relation is

$$y(k) - y(k{-}1) = M\big(h(k)\big)\,\big(u(k) - u(k{-}1)\big), \tag{5.17}$$

with the history at sample k

$$h(k) = \big|u(k) - u(\bar{k})\big|, \tag{5.18}$$

and $\bar{k}$ the most recent reversal sample. The parameters of $\hat{M}$ are fitted to the proxy by least squares,

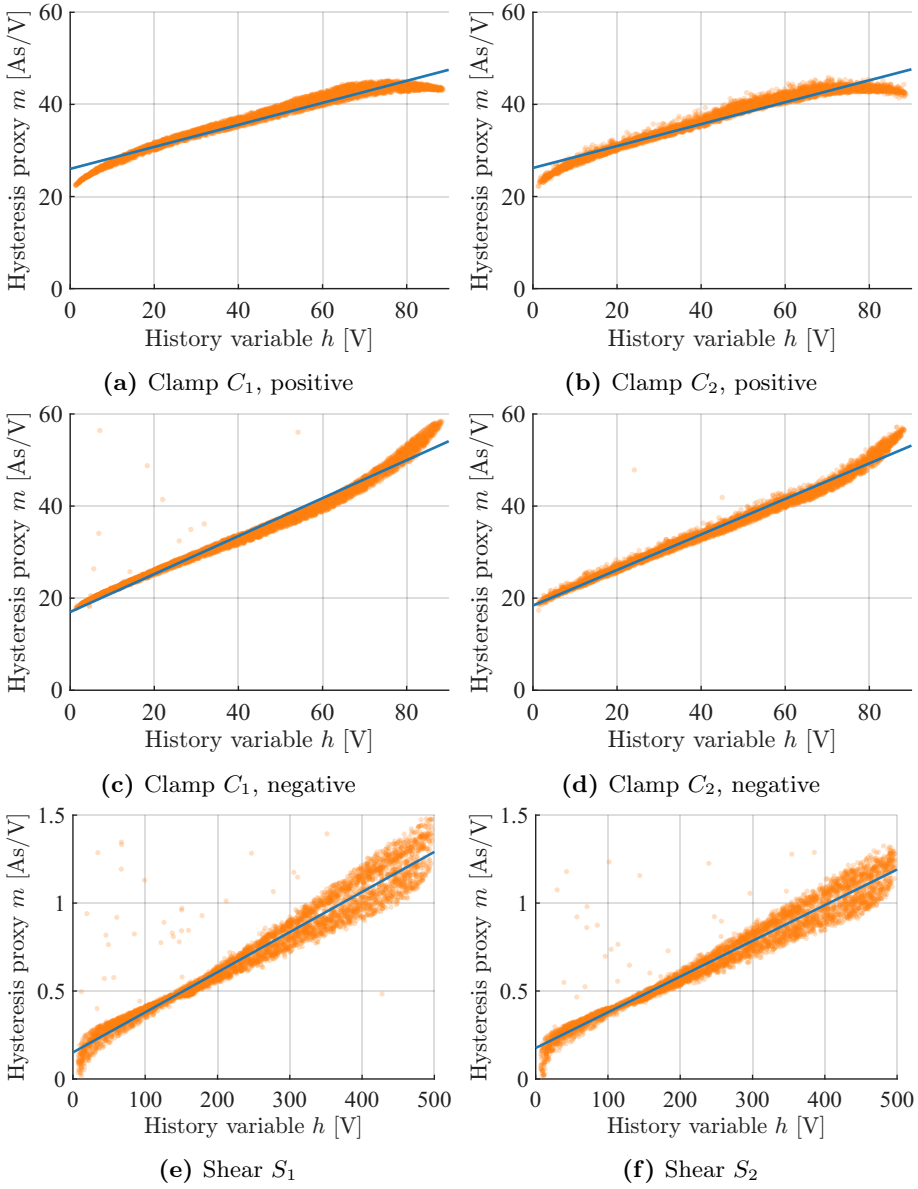
$$(\hat{\theta}_1, \hat{\theta}_2) = \arg\min_{(\theta_1,\,\theta_2)\,\in\,\mathbb{R}^2} \sum_{k=1}^{N_M} \big(m(k) - \theta_1\, h(k) - \theta_2\big)^2, \tag{5.19}$$

over the N_M recorded samples, leaving out those near a reversal where $\dot{u} \approx 0$ makes $m(k)$ unreliable. Following the clamp and shear split above, each clamp element is fitted separately for its two stepping directions and each shear element with a single parameter pair.

Inverting (5.17) and using the identified $\hat{M}$ in place of M yields the feedforward control law

$$u(k) = u(k{-}1) + \frac{r(k) - r(k{-}1)}{\hat{M}\big(h(k{-}1)\big)}, \tag{5.20}$$

where $r(k)$ is the reference displacement from the (possibly ILC-modified) waveform. The history variable at the previous sample $h(k-1)$ is used in place of $h(k)$, since $h(k)$ depends on the voltage $u(k)$ before it is applied.



(a) Clamp C_1, positive **(b)** Clamp C_2, positive

(c) Clamp C_1, negative **(d)** Clamp C_2, negative

(e) Shear S_1 **(f)** Shear S_2

Figure 5.7. Measured hysteresis proxy m (5.15) against the history variable h (5.18) for the clamp and shear elements, each pooling samples over a range of drive frequencies (●), with the affine fit (5.16) (—). Panels **(a)**–**(d)** show the clamps for both stepping directions, and panels **(e)**–**(f)** show the shears.

The proxy (5.15) identifies $\hat{M}$ only up to the unknown constant ξ, so the control law (5.20) reaches the intended displacement up to that same scale. This scale is inconsequential in practice: the reference r carries the same units, and its amplitude is set so that the applied voltage spans the full element range over a commutation step, which absorbs ξ into the reference [57].

5.4.2 ILC for α-Domain Disturbance Compensation

In van Meer et al. [57], a single compensation function was learned for one PSA, using the mover position error from a capacitive sensor. This section extends the approach to three PSAs by learning compensation functions sequentially: each PSA is calibrated individually while the others remain stationary, so that the POI error can be unambiguously attributed to the active PSA. The POI error is obtained from the high-rate proxy $\hat{p}$ (Section 5.3.3) and projected onto piezo coordinates through the inverse kinematics.

The ILC correction modifies the shear waveforms only; clamp waveforms control the contact timing and are not adjusted by ILC. Because only one shear element is in contact with the mover at any time, a single scalar compensation function simultaneously modifies both shear waveforms. Separate functions are learned for positive and negative stepping directions, giving six compensation functions in total across the three PSAs.

During calibration of PSA i, only that PSA moves while the other two remain stationary. The ILC uses the projected error ε_i of (5.3), evaluated at the POI proxy $p_e = \hat{p}$. Because only PSA i is moving, this error isolates the contribution of that PSA, while the errors in the other two coordinates remain near zero (aside from measurement noise and kinematic coupling) and are not used for learning. This sequential approach simplifies the calibration procedure and ensures that each compensation function addresses only the disturbances from its own PSA.

As in Section 5.4.1, the learning procedure is identical for every compensation function, so the equations below are written for a single generic case: PSA and direction indices are omitted and ε denotes the projected error ε_i for whichever PSA is under consideration. The ILC is performed in open-loop fashion. The piezo walks at a constant rate through the commutation cycle, and the measurements are used only to update the feedforward compensation for the next trial.

The modified waveform is

$$\tilde{\rho}(\alpha) = \rho(\alpha) + f(\alpha), \tag{5.21}$$

where ρ is the nominal waveform and f is the learned correction. Because the correction is parameterized in α rather than in time, a correction learned at one stepping speed applies at any other speed for which the stage behaves quasi-statically. When these modified waveforms are used together with the hysteresis

compensation of Section 5.4.1, the resulting element displacements approximately track the modified references and the mover follows the intended trajectory more accurately.

The update law for trial j reads

$$f_{j+1}(k) = Q(z)\big(f_j(k) + L(z)\,\varepsilon_j(k)\big), \tag{5.22}$$

with robustness filter Q and learning filter L, and where G denotes the plant transfer from element waveform modification f to the projected PSA error ε. Substituting (5.22) into the system yields the trial-to-trial error dynamics $\varepsilon_{j+1} = Q(1 - LG)\,\varepsilon_j$ [130]. With $L = G^{-1}$ and $Q = 1$, the factor $(1 - LG)$ vanishes, so $\varepsilon_{j+1} = 0$ regardless of the initial error and perfect compensation is achieved in a single trial. In practice, however, G is not perfectly known and L can only approximate G^{-1}, so that Q must be designed to ensure convergence despite the model mismatch.

A sufficient condition for monotonic convergence of (5.22) is [130]

$$\sup_{\omega\in[0,\pi]} \big|Q(e^{j\omega})\big(1 - L(e^{j\omega})\,G(e^{j\omega})\big)\big| < 1. \tag{5.23}$$

The learning filter is set to $L = \hat{G}^{-1}$, where $\hat{G}$ is a parametric model of G (identified below). Ideally, $|Q|$ should be close to 1 everywhere to retain the full learned correction. However, at frequencies where $\hat{G}$ deviates from G, the factor $(1 - LG)$ is no longer small and $|Q| = 1$ can violate (5.23). In many applications, the model is most accurate at low frequencies, so Q is designed as a low-pass Butterworth filter. The convergence condition is evaluated on a non-parametric frequency response $\bar{G}(e^{j\omega})$ obtained from measured data (see below) to more closely reflect the true system behavior.

The shear element plant model is identified following van Meer et al. [57] using a best-linear-approximation (BLA) procedure [135]: the commutation angle α is perturbed by a random-phase multisine signal, and the resulting displacement is measured via the corrected encoder estimate $\hat{p}$. The BLA extracts a non-parametric frequency response $\bar{G}(e^{j\omega})$ from this input–output data, to which a low-order parametric model $\hat{G}$ is fitted.

Remark 5.1. *The ILC operates in piezo coordinates, since both the projected error ε_i and the waveform correction f live in the q-domain. The plant model $\hat{G}$ must therefore be identified consistently with the sensor-deviation-corrected output used for learning. The recorded output is the projected corrected estimate $[\Gamma(\hat{p}, \varphi)]_i$ rather than the raw encoder reading q_i, so that $\hat{G}$ captures the transfer from waveform modification to projected corrected error. The two outputs differ by $\Delta_i(\alpha_i)$, a static function of the commutation angle α, which is the excitation variable. A static (nonlinear) map contributes at most a real, frequency-independent constant*

to the best linear approximation, and its remaining effect is a stochastic nonlinear distortion that only raises the variance of $\bar{G}$ [135], so Δ_i introduces no frequency-dependent dynamics into $\hat{G}$. Since the same corrected output is used both for identifying $\hat{G}$ and for running the ILC, this constant is consistently absorbed into $L = \hat{G}^{-1}$, and the convergence condition (5.23) is unaffected.

The time-domain signal $f_{j+1}(k)$ from (5.22) must be converted to a function of α. The compensation function is parameterized as a linear combination of basis functions,

$$f(\alpha) = \boldsymbol{\psi}_f^\top(\alpha)\,\boldsymbol{\gamma}, \tag{5.24}$$

where $\boldsymbol{\psi}_f(\alpha) \in \mathbb{R}^{n_f}$ collects n_f basis functions and $\boldsymbol{\gamma} \in \mathbb{R}^{n_f}$ are the coefficients. In practice, piecewise-linear basis functions on n_f equidistant grid points in $[0, 2\pi)$ are used. After each ILC update (5.22), the time-domain signal is projected onto this basis by minimizing the weighted squared residual

$$\boldsymbol{\gamma}_{j+1} = \arg\min_{\boldsymbol{\gamma}\in\mathbb{R}^{n_f}} \left\|\hat{\mathbf{G}}(\boldsymbol{\Psi}_f\boldsymbol{\gamma} - \mathbf{f}_{j+1})\right\|_2^2, \tag{5.25}$$

where $\boldsymbol{\Psi}_f = [\boldsymbol{\psi}_f(\alpha(1)), \ldots, \boldsymbol{\psi}_f(\alpha(N))]^\top$ stacks the basis function evaluations, $\hat{\mathbf{G}}$ is the lifted impulse response matrix of $\hat{G}$, and $\mathbf{f}_{j+1}$ stacks the time-domain signal $f_{j+1}(k)$, see van Meer et al. [57]. The minimization has the same structure as (5.12), with the solution

$$\boldsymbol{\gamma}_{j+1} = (\boldsymbol{\Psi}_f^\top \hat{\mathbf{G}}^\top \hat{\mathbf{G}} \boldsymbol{\Psi}_f)^{-1} \boldsymbol{\Psi}_f^\top \hat{\mathbf{G}}^\top \hat{\mathbf{G}}\, \mathbf{f}_{j+1}. \tag{5.26}$$

The premultiplication by $\hat{\mathbf{G}}$ ensures that the projection minimizes the predicted output error rather than the compensation signal itself. The basis parameterization and projection modify the convergence analysis. The exact necessary and sufficient condition for monotonic convergence of the combined scheme (5.22)–(5.26) involves both $\boldsymbol{\Psi}_f$ and $\hat{\mathbf{G}}$ [57, Lemma 4.1], [136]. However, van Meer et al. [57, Theorem 4.2] shows that the frequency-domain condition (5.23) is sufficient regardless of the basis choice, so that condition is used here for filter design. Figure 5.8 verifies (5.23) on the measured FRF $\bar{G}$ of each PSA, shown here for the representative one. The cutoff of Q is designed to be as high as possible while maintaining $|Q(1 - L\bar{G})| < 1$.

5.5 Experimental Results

This section evaluates the framework experimentally, first on the lab setup and then on the operational electron microscope. All displacement quantities in the results below are reported in arbitrary units (a.u.), obtained through a single fixed scaling applied consistently across every figure and value.

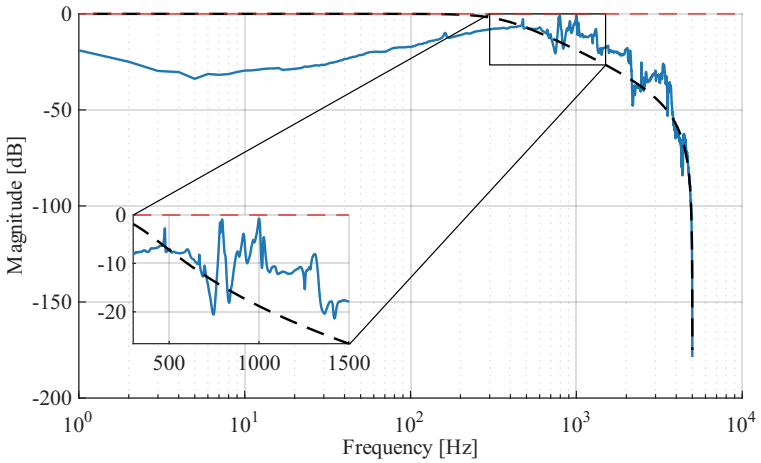


Figure 5.8. Verification of the ILC convergence condition (5.23) for the representative PSA. Shown are $|Q(1 - L\bar{G})|$ (——), $|Q|$ (- - -), and the 0 dB threshold (- - -). The inset zooms the region around 1 kHz where the margin is smallest; the condition is satisfied across the full frequency range.

Throughout, the tracking error is evaluated in piezo coordinates. Only one PSA is stepped at a time, so its error is a single scalar, obtained by projecting a position signal onto that PSA's coordinate through the inverse kinematics and subtracting the piezo reference, as in (5.3). We report this scalar for three position signals, namely the raw encoder estimate $\hat{p}_0$, the encoder estimate after sensor-deviation correction $\hat{p}$, and the true POI measurement given by the capacitive reading p_c on the lab setup and the vision-based estimate p_s on the microscope. On the lab setup, the framework was calibrated on all three PSAs and the improvements are comparable across them. The results below therefore focus on one representative PSA on both the lab setup and the microscope. On the EM, its axis is aligned with the imaging plane by choice of tilt angle (Section 5.3.1), so the in-plane image shift observes the full motion of that axis.

Four feedforward strategies are compared throughout the experiments. The difference between them lies in which compensators are active:

- **S1** Traditional feedforward with a constant (non-compensated) hysteresis model and the unmodified nominal waveforms ρ, i.e., $\hat{M}(h) = 1$, $\Delta(\alpha) = 0$, and $f(\alpha) = 0$.
- **S2** Rate-independent hysteresis compensation only (Section 5.4.1), with $\Delta(\alpha) = 0$ and $f(\alpha) = 0$.
- **S3** Hysteresis compensation and ILC, without sensor deviation correction: $\Delta(\alpha) = 0$.

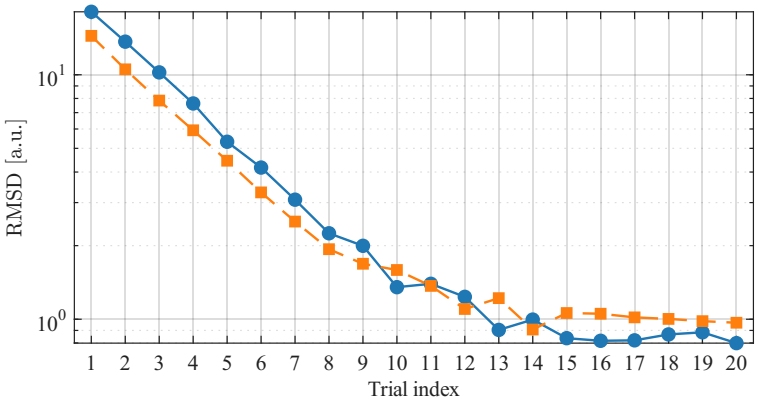


Figure 5.9. RMSD of the projected tracking error ε_i (5.3) versus ILC trial index. Forward stepping (—●—) and backward stepping (- ■ -) are overlaid. Hysteresis compensation is active in all trials.

S4 Full framework: hysteresis compensation, sensor deviation correction, and ILC.

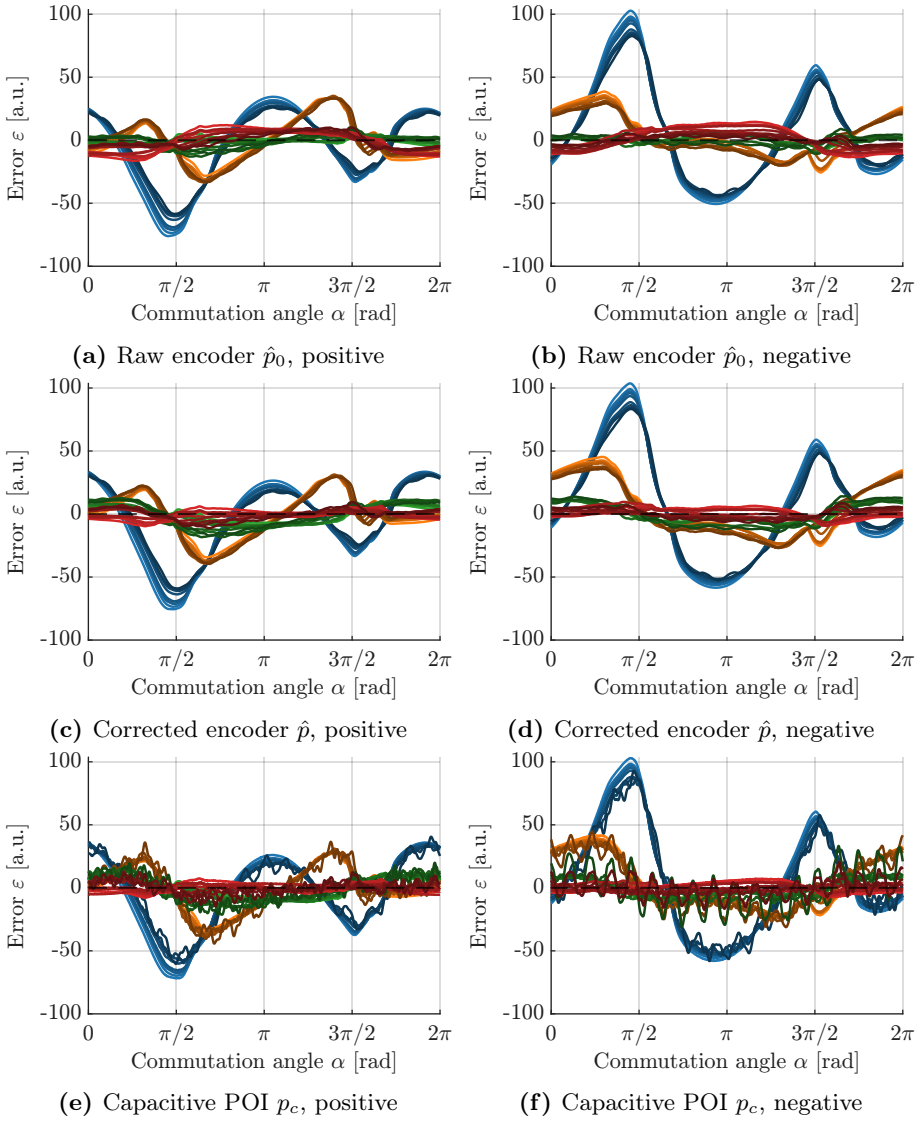
5.5.1 ILC Convergence

The ILC compensation is learned at a drive frequency of 2 Hz, on the lab setup for each PSA in turn and on the EM for the representative PSA. Because the correction is parameterized in α, it is subsequently applied across the full tested frequency range. Figure 5.9 shows the RMSD of the projected tracking error ε_i (5.3) over successive ILC trials for the representative PSA on the lab setup, in both stepping directions. Hysteresis compensation is active in all trials.

5.5.2 Positioning Performance on the Lab Setup

Figure 5.10 compares the POI tracking error versus commutation angle for strategies S1, S2, S3, and S4 on the lab setup at drive frequencies from 0.1 Hz to 10 Hz. Figure 5.11 summarizes the error per drive frequency.

From the figures, each feedforward component contributes to improved POI tracking. At a drive frequency of 1 Hz, the POI RMSD measured by the capacitive sensor drops from 35.5 a.u. for the traditional strategy S1 to 2.7 a.u. for the full framework S4 in the positive stepping direction, and from 48.5 a.u. to 3.5 a.u. in the negative direction. These are 13- and 14-fold reductions, respectively, and the same trend holds across the tested drive frequencies (Figure 5.11). At the highest tested drive frequencies, the performance of S3 and S4 drops off slightly, which



(a) Raw encoder $\hat{p}_0$, positive

(b) Raw encoder $\hat{p}_0$, negative

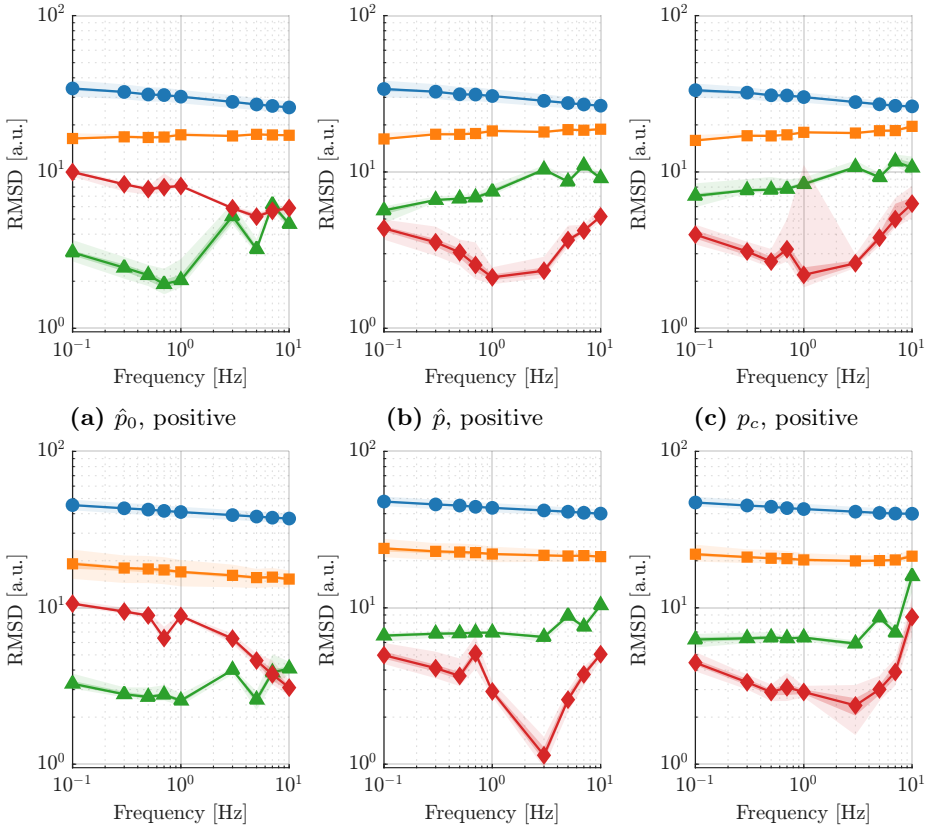
(c) Corrected encoder $\hat{p}$, positive

(d) Corrected encoder $\hat{p}$, negative

(e) Capacitive POI p_c, positive

(f) Capacitive POI p_c, negative

Figure 5.10. Mean projected tracking error ε_i (5.3) versus commutation angle α_i on the lab setup. The left column is the positive stepping direction and the right column the negative direction, with the three position estimates $\hat{p}_0$, $\hat{p}$, and p_c on successive rows. Each panel shows strategies S1 (—), S2 (—), S3 (—), and S4 (—), with lines darkening from 0.1 Hz (lightest) to 10 Hz (darkest).



(a) $\hat{p}_0$, positive **(b)** $\hat{p}$, positive **(c)** p_c, positive

(d) $\hat{p}_0$, negative **(e)** $\hat{p}$, negative **(f)** p_c, negative

Figure 5.11. RMSD of the projected tracking error ε_i (5.3) versus drive frequency: S1 (—●—), S2 (—■—), S3 (—▲—), S4 (—◆—); shaded bands of matching color denote the IQR and 5th–95th percentile range over repeated steps. The three position estimates are the raw encoder $\hat{p}_0$, the sensor-deviation-corrected encoder $\hat{p}$, and the capacitive POI p_c.

is mostly attributed to flexible-mode behavior of the stage that the quasi-static feedforward does not target.

Figure 5.10 also shows that the raw encoder error for strategy S4 is worse than for S3. This is by design: the sensor deviation correction in combination with ILC results in a shaped non-zero error at the encoder level, but it is precisely this shaped error that yields reduced error at the POI, which is the actual control objective.

5.5.3 Positioning Performance on the EM

The lab-setup results above confirm the feedforward components (hysteresis compensation, sensor deviation, and ILC) using capacitive ground truth at the POI. On the EM, the feedforward models are re-identified following the same procedures as on the lab setup, with hysteresis from element voltage and current, sensor deviation from p_s, and ILC compensation from the corrected encoder estimate $\hat{p}$.

Figure 5.12 shows the projected tracking error ε_i as a function of commutation angle α_i for a representative PSA on the EM stage at drive frequencies from 0.5 Hz to 2 Hz, analogous to Figure 5.10 for the lab setup. The residual structure on the EM mirrors that on the lab setup, confirming that the same physical effects are present and that the modeling approach transfers to the operational microscope. Figure 5.13 summarizes the EM tracking error versus drive frequency, mirroring Figure 5.11 for the lab setup. At 1 Hz, the full framework S4 reduces the POI RMSD from 32.7 a.u. to 4.7 a.u. in the positive stepping direction, and from 55.2 a.u. to 4.7 a.u. in the negative direction, a 7-fold and 12-fold reduction. The improvement is consistent across the tested drive frequencies. We observe a slight performance drop compared to the lab setup, only in the ILC contribution. This is attributed to position-dependent stage dynamics: the ILC compensation is learned at the stage home position, whereas the vision-based validation requires imaging the specimen at a different stage position.

5.6 Conclusion

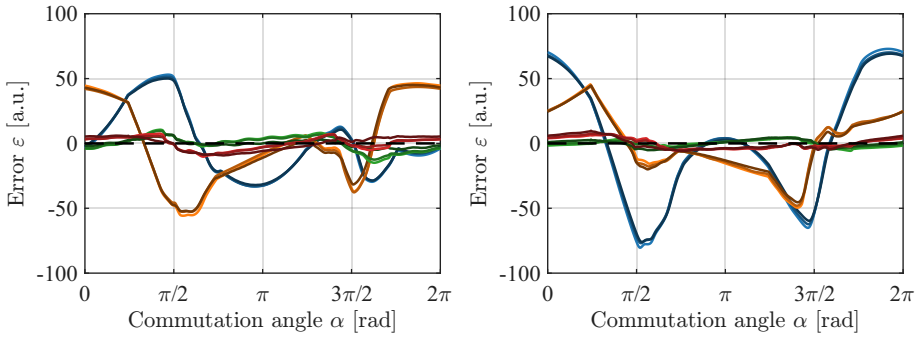
This chapter presented an integrated framework for precision specimen positioning in an operational electron microscope. Positioning accuracy in such instruments is limited by hysteresis, mechanical misalignments, non-collocated sensing, and non-repeatable disturbances such as thermal drift. This work builds upon prior work that has addressed several of these challenges on a simplified lab setup [57]. In this work the framework is extended to operation in a real transmission electron microscope.

The gap between encoder and POI coordinates is bridged by a commutation-angle-dependent sensor deviation that is physically motivated by bending from clamp-shear contact forces. The deviation is learned from whichever ground-truth sensor is available: capacitive sensors p_c on the lab setup, accumulated image-tracker shifts p_s on the EM. The corrected encoder estimate $\hat{p}$ provides a high-rate POI proxy that drives feedforward learning in both environments. The α-domain ILC is extended from one PSA to three by projecting POI errors onto per-PSA signals through the inverse kinematics Γ. The hysteresis model is simplified from the rate-dependent radial-basis-function formulation used in van Meer et al. [57]. The new model is an affine function of the history variable, which reduces the

parameter count from the number of radial basis functions to two per element.

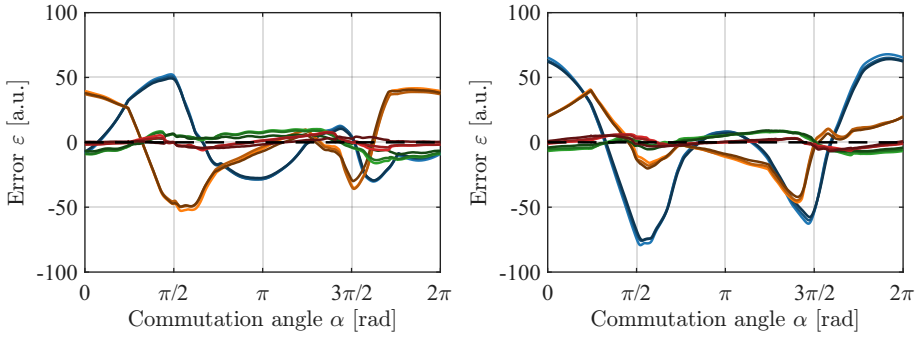
On the lab setup, the framework reduces the POI RMSD 13-fold (positive stepping) and 14-fold (negative stepping) at 1 Hz, and on the electron microscope 7-fold and 12-fold, respectively. The improvement is consistent across the tested drive frequency range, though at higher drive frequencies performance slightly degrades, likely due to flexible-mode behavior. Importantly, this accuracy is reached with no sensor at the POI, since the corrections are learned from the images that the microscope already acquires. The framework therefore enables nanometer-scale specimen positioning without added sensing hardware.

Several directions remain for future work. Firstly, the performance gain during tomographic tilt series should be validated in practice. Next, combining the present feedforward framework with the image-based feedback of van Horssen et al. [56] would let the two together reject both the repeatable and the non-repeatable disturbances. This becomes more powerful once the image and encoder measurements are fused: the low-level feedback could then be driven by a fused, high-rate POI estimate rather than the encoder signal alone, which has the potential to substantially improve closed-loop performance. Estimating the out-of-plane coordinate p_z from image defocus would extend the vision sensor to three dimensions. Finally, flexible-mode vibrations are repeatable but not α-periodic, so the current feedforward does not address them; time-domain approaches such as input shaping could target these residuals.



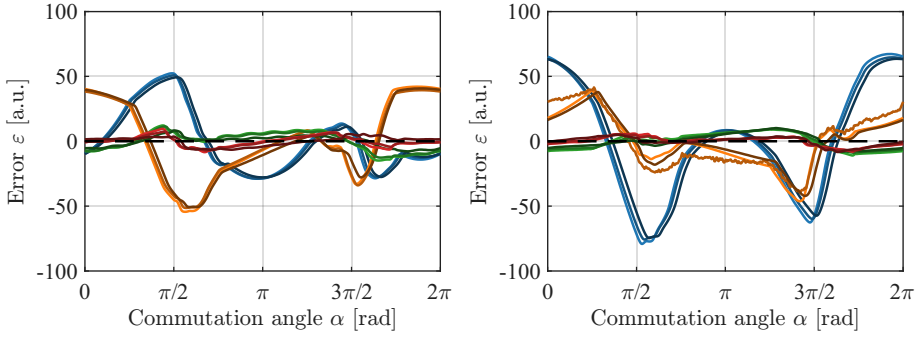
(a) Raw encoder $\hat{p}_0$, positive

(b) Raw encoder $\hat{p}_0$, negative



(c) Corrected encoder $\hat{p}$, positive

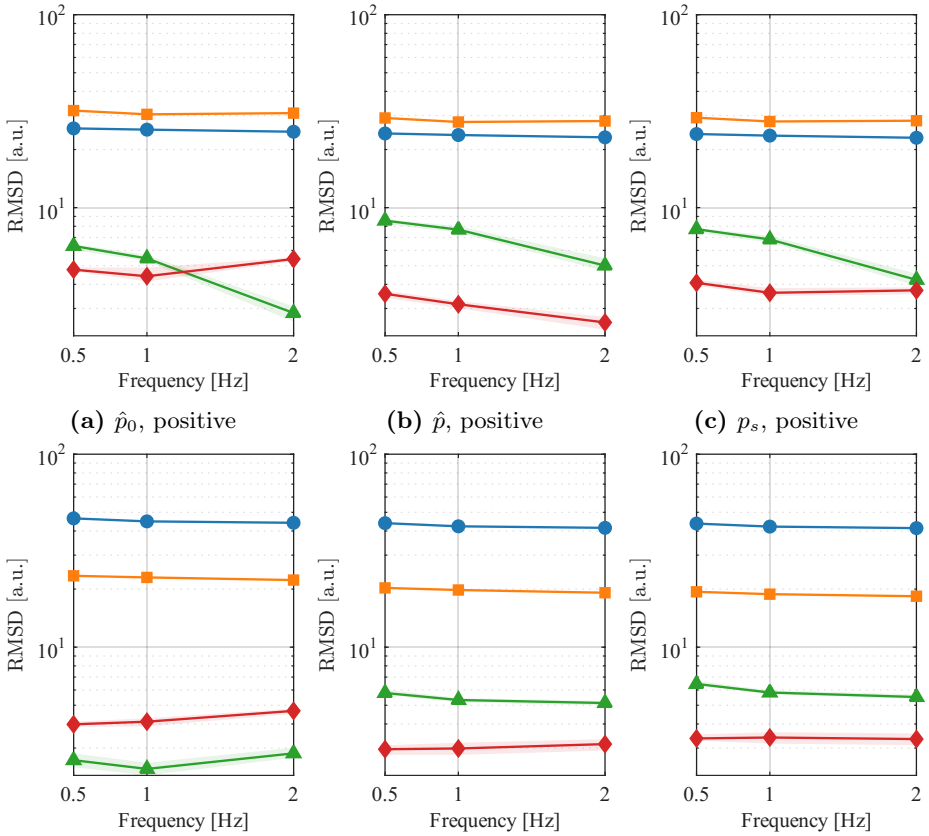
(d) Corrected encoder $\hat{p}$, negative



(e) Vision-based POI p_s, positive

(f) Vision-based POI p_s, negative

Figure 5.12. Mean projected tracking error ε_i (5.3) versus commutation angle α_i on the EM stage. The left column is the positive stepping direction and the right column the negative direction, with the three position estimates $\hat{p}_0$, $\hat{p}$, and p_s on successive rows. Each panel shows strategies S1 (—), S2 (—), S3 (—), and S4 (—), with lines darkening from 0.5 Hz (lightest) to 2 Hz (darkest).



(a) $\hat{p}_0$, positive **(b)** $\hat{p}$, positive **(c)** p_s, positive

(d) $\hat{p}_0$, negative **(e)** $\hat{p}$, negative **(f)** p_s, negative

Figure 5.13. RMSD of the projected tracking error ε_i (5.3) versus drive frequency on the EM stage: S1 (—●—), S2 (—■—), S3 (—▲—), S4 (—◆—); shaded bands of matching color denote the IQR and 5th–95th percentile range over repeated steps. The three position estimates are the raw encoder $\hat{p}_0$, the sensor-deviation-corrected encoder $\hat{p}$, and the vision-based POI p_s.

III

Data-Efficient Machine Learning

6

Constructing VAE Latent Spaces with Prescribed Topology

Abstract - Variational autoencoders (VAEs) learn low-dimensional latent representations of high-dimensional data. When the data lies on a manifold with non-Euclidean topology, the standard Gaussian prior introduces a topological mismatch that degrades reconstruction quality and prevents faithful representation. We present a constructive mathematical framework that resolves this mismatch for all manifolds that admit a product covering space. These are manifolds expressible as products of elementary factors (circles, intervals, lines, or hyperspheres) or as quotients of such products by a finite symmetry group. The class includes cylinders, tori, Möbius strips, Klein bottles, and real projective spaces. Factorized distributions over the elementary factors yield product topologies with closed-form, decoupled KL divergences, so that each latent factor can be shaped independently while keeping training tractable. We catalog reparametrizable encoder-prior pairs for periodic, bounded, and unbounded supports, and provide coordinate transformations that allow standard neural networks to output non-Euclidean parameters with smooth gradients. For quotient manifolds, the decoder receives group-invariant features of the covering-space coordinates, so that identified points produce identical outputs. Anchor constraints fix the coordinate system relative to the data or create soft topological holes. Experiments on synthetic manifolds and real-image datasets (rotated and cyclically shifted

This chapter is based on van Hulst et al. [137].

MNIST) confirm that a topology-matched prior aligns KL regularization with the data manifold. The resulting topology-aware models preserve the intrinsic distance structure of the data manifold far better than the Gaussian baseline at matched KL divergence, and reconstruct more accurately whenever regularization is strong enough to shape the representation. The code is available at `https://github.com/JvHulst/VAE-Topology`.

6.1 Introduction

Variational autoencoders (VAEs), introduced by Kingma and Welling [89], have become a fundamental tool for learning low-dimensional representations of high-dimensional data. The main innovation of VAEs is the *reparametrization trick* [138], which expresses latent samples as deterministic transformations of encoder outputs and independent noise, enabling gradient-based optimization of the loss criterion.

A key motivation for dimensionality reduction is that many real-world datasets, despite their high-dimensional representation, lie on or near a low-dimensional manifold. In such cases, we would like the learned latent space to faithfully represent the structure of this underlying manifold. However, the standard VAE framework assumes a Gaussian prior, which induces a Euclidean topology on the latent space. This creates a fundamental mismatch when the data manifold has non-Euclidean structure, for instance, when data lies on a sphere, torus, or bounded domain.

This mismatch has motivated several approaches to treat non-Euclidean latent spaces. Davidson et al. [139] introduced the hyperspherical VAE, which uses the von Mises-Fisher distribution for latent spaces on hyperspheres S^{n-1}. A follow-up [140] scaled this to higher dimensions using a product of vMF distributions. Falorsi et al. [141] extended this to general Lie groups such as $SO(3)$, providing a principled approach for rotational latent spaces. Mathieu et al. [142] proposed hyperbolic VAEs, where the Poincaré disc latent space is well-suited to data with hierarchical structure. Kalatzis et al. [143] extended this direction by learning Riemannian metric tensors to capture the geometry of learned representations. Perez Rey et al. [144] and Nagano et al. [145] developed diffusion-based and wrapped normal distributions for VAEs on tori, spheres, and hyperbolic spaces. Normalizing flows on manifolds [146–148] take a different approach: rather than prescribing the prior topology, they compose diffeomorphisms to learn a flexible distribution, at the cost of Jacobian determinants and Monte Carlo KL estimation. Equivariant representations [149] impose structure through architectural constraints, while topological autoencoders [150] use persistent homology to preserve topological structure in learned representations. Tomczak and Welling [151] showed more generally that the choice of prior distribution has a key effect on the quality of learned representations, even without changing the latent topology. Miao et al. [152] proposed an alternative mechanism: rather than modifying the prior, they keep a Gaussian prior and impose inductive biases through a learned parametric mapping from an intermediary latent space. Although these works have substantially advanced non-Euclidean representation learning, none provides a constructive design procedure for VAEs with a prescribed arbitrary topology while retaining a tractable closed-form ELBO. In this chapter, we develop a unified framework that covers a broad class of manifolds, including many that arise in practice, and provides a systematic pipeline for constructing the corresponding VAEs.

The specific class of manifolds that we consider are those that admit a *product covering space*: manifolds that can be expressed as a product of elementary factors (circles S^1, bounded intervals $[0, 1]$, half-lines $[0, \infty)$, real lines $\mathbb{R}$, or hyperspheres S^{n-1}), or as a quotient of such a product by a finite symmetry group G. This class includes cylinders, tori, annuli, and hypercubes as product manifolds, and Möbius strips, Klein bottles, and real projective spaces as finite quotients of products. Product topologies arise naturally when data has periodic or bounded factors of variation. For example, dihedral angles in protein backbone geometry live on tori [153, 154], and joint angles in robotics produce similar toroidal configuration spaces [155]. Quotient topologies arise when a discrete symmetry identifies points in the product space. For instance, non-orientable topologies such as Möbius strips and Klein bottles arise in the analysis of natural image patches [156], and real projective spaces describe the orientations of objects with head-to-tail symmetry [157].

Our main contribution is a constructive framework that provides topology-aware VAEs with tractable ELBO objectives and end-to-end gradient-based training for any manifold in this class. Based on this result, we provide a systematic pipeline for building the corresponding VAE. The pipeline consists of the following steps:

1. Express $\mathcal{M}$ as a product of elementary factors, or identify a product covering space $\widetilde{\mathcal{Z}}$ such that $\mathcal{M} = \widetilde{\mathcal{Z}}/G$. We prove that factorized encoder-prior distributions over the factors yield product topologies with closed-form, decoupled KL divergences (Propositions 6.1–6.2).

2. For each factor, select an encoder-prior distribution pair from a catalog of reparametrizable distributions with tractable KL divergences (Table 6.1). We provide pairs for periodic, bounded, and unbounded supports, together with coordinate transformations χ that allow standard neural networks to output the required non-Euclidean parameters with smooth gradients.

3. For quotient manifolds $\mathcal{M} = \widetilde{\mathcal{Z}}/G$, construct a G-invariant feature map ρ that embeds latent coordinates in Euclidean space for the decoder, ensuring that identified points produce identical decoder outputs. We prove that such maps exist whenever G is finite (Proposition 6.3) and give explicit constructions for Möbius strips and Klein bottles.

4. Add anchor constraints that fix the coordinate system relative to the data by pinning known observations to prescribed latent locations, or, in repulsive form, create soft topological holes.

Figure 6.1 illustrates this pipeline with a Möbius strip example. The framework preserves the usual VAE benefits: the KL term encourages smooth, well-covered representations within the chosen topology, and the reparametrization trick enables end-to-end gradient-based training. In Section 6.4, we validate this pipeline on

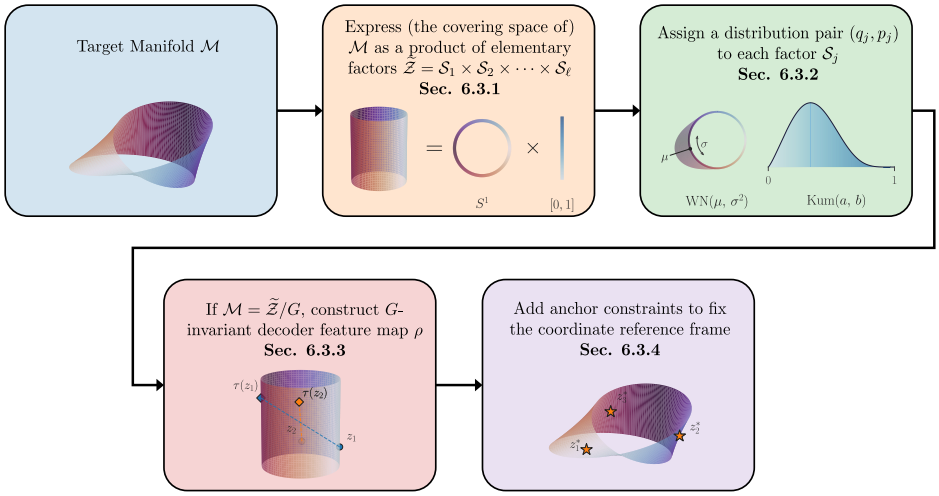


Figure 6.1. Pipeline for designing a topology-aware VAE, illustrated with the running example of a Möbius strip. **Step 1:** Given a target manifold $\mathcal{M}$, find a product covering space $\widetilde{\mathcal{Z}}$ and decompose it into elementary factors. The Möbius strip is the quotient of a cylinder by a $\mathbb{Z}_2$ action, so $\widetilde{\mathcal{Z}} = S^1 \times [0, 1]$. **Step 2:** Assign a compatible encoder–prior pair from Table 6.1 to each factor. Here, a Wrapped Normal is assigned to S^1 and a Kumaraswamy to $[0, 1]$. **Step 3:** If $\mathcal{M}$ is a quotient, construct a G-invariant feature map ρ so that the decoder receives only features that respect the identification; for the Möbius strip, this ensures $(h, \theta) \sim (1 - h, \theta + \pi)$ produce identical outputs. When $\mathcal{M}$ is already a product, $\widetilde{\mathcal{Z}} = \mathcal{M}$ and this step is trivial ($\rho(z) = z$). **Step 4:** Add anchor constraints to fix the reference frame.

three synthetic manifolds (cylinder, torus, Möbius strip) with known ground truth, and on two real-image datasets (rotated MNIST and cyclically shifted MNIST). We show that when the prior matches the data topology and the coordinate system is grounded through anchoring, the KL term regularizes the latent representation within the correct manifold rather than against it. Topology-aware models then preserve the distance structure of the data manifold substantially better than the Gaussian baseline, and reconstruct more accurately across the range of regularization strengths in which the KL term meaningfully shapes the representation.

The remainder of this chapter is organized as follows. Section 6.2 formalizes the topology mismatch problem and reviews the VAE framework. Section 6.3 develops the four tools in the pipeline: product topologies, distribution pairs, quotient topologies, and anchor constraints. Section 6.4 presents experimental validation on

three synthetic datasets with known manifold structure, followed by two real-image experiments. Section 6.5 discusses conclusions and future directions.

6.2 Problem Formulation

We begin by formalizing the relationship between data manifolds and latent space topology, then review the VAE framework and identify the mathematical requirements for valid inference with non-standard priors.

6.2.1 Data on Low-Dimensional Manifolds

Many modern data analysis problems involve finding structure in high-dimensional observations that lie on or near a low-dimensional *manifold* [158], a space that locally resembles $\mathbb{R}^n$ but may have a different *topology*, or global shape. For instance, a circle S^1 is locally like $\mathbb{R}$ but has a periodic topology that $\mathbb{R}$ does not.

We now formalize the data-generating setting. Let $\mathcal{M}$ be an n-dimensional manifold representing the space of underlying factors of variation. High-dimensional observations $y \in \mathbb{R}^p$ (with $n \ll p$) are generated via

$$y = f(x) + \eta, \tag{6.1}$$

where $x \in \mathcal{M}$ is a point on the manifold, $f : \mathcal{M} \to \mathbb{R}^p$ is the generative function, and η represents observation noise. We call x the *true latent variable* associated with observation y, and $\mathcal{M}$ the *true latent space*. We assume f is *injective* on $\mathcal{M}$, meaning that distinct true latent variables produce distinguishable observations. When a generative function is not injective, one can sometimes restore injectivity by passing to the quotient space $\mathcal{M}/\sim$, where $x_1 \sim x_2$ if and only if $f(x_1) = f(x_2)$.

Given data generated according to (6.1), two natural goals arise. The first is *dimensionality reduction*: finding a low-dimensional representation that captures the essential structure of $\mathcal{M}$. The second is *generative modeling*: learning to generate new observations consistent with (6.1). VAEs are a tool for achieving both goals simultaneously. However, when $\mathcal{M}$ has a nontrivial topology, standard VAEs face limitations that we address in Section 6.2.3.

6.2.2 Variational Autoencoders

A VAE [89] introduces a *latent space* $\mathcal{Z}$ of dimension ℓ and represents each observation y by a *latent variable* $z \in \mathcal{Z}$, learned through an encoder-decoder architecture trained via variational inference. The VAE has three components: an *encoder* $q_\phi(z|y)$, a distribution over $\mathcal{Z}$ parameterized by trainable weights ϕ; a *decoder* $p_\psi(y|z)$, a distribution over observation space $\mathbb{R}^p$ parameterized by weights ψ; and a *prior* $p(z)$ over the latent variables. Figure 6.2 illustrates the architecture.

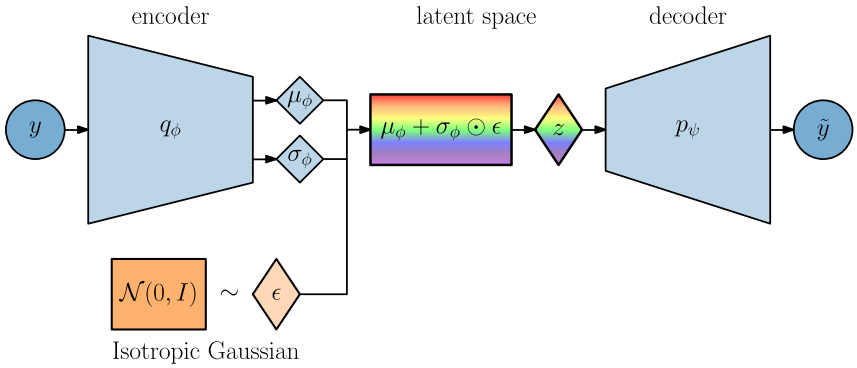


Figure 6.2. Standard VAE architecture. The encoder networks μ_ϕ and σ_ϕ produce the parameters of the encoder distribution $q_\phi(z|y)$. A latent sample is drawn via the reparametrization trick: $z = \mu_\phi(y) + \sigma_\phi(y) \odot \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$. The decoder network μ_ψ maps z to a reconstruction $\hat{y} = \mu_\psi(z)$.

In the standard VAE, the encoder distribution is a diagonal Gaussian whose mean $\mu_\phi : \mathbb{R}^p \to \mathbb{R}^\ell$ and standard deviation $\sigma_\phi : \mathbb{R}^p \to \mathbb{R}^\ell_{>0}$ are produced by neural networks:

$$q_\phi(z|y) = \mathcal{N}(z;\ \mu_\phi(y),\ \mathrm{diag}(\sigma^2_\phi(y))). \tag{6.2}$$

The decoder distribution is also Gaussian, with mean $\mu_\psi : \mathbb{R}^\ell \to \mathbb{R}^p$ produced by a neural network and a scalar variance σ^2_ψ that is either fixed or learned:

$$p_\psi(y|z) = \mathcal{N}(y;\ \mu_\psi(z),\ \sigma^2_\psi I). \tag{6.3}$$

The decoder mean $\hat{y} = \mu_\psi(z)$ is the reconstruction of y, and σ^2_ψ controls the assumed observation noise level. The standard prior is a zero-mean Gaussian with identity covariance (i.e., isotropic): $p(z) = \mathcal{N}(0, I)$.

The VAE is trained by maximizing the evidence lower bound (ELBO) over a dataset $\{y_i\}_{i=1}^N$. For a single observation y, the ELBO is

$$\mathcal{L}(\phi, \psi; y) = \underbrace{\mathbb{E}_{q_\phi(z|y)}[\log p_\psi(y|z)]}_{\text{Reconstruction}} - \beta \underbrace{\mathrm{KL}(q_\phi(z|y) \| p(z))}_{\text{Regularization}}, \tag{6.4}$$

where $\beta > 0$ is a scalar weight. The Kullback-Leibler (KL) divergence [159] between two distributions q and p is defined as

$$\mathrm{KL}(q\|p) = \mathbb{E}_{z\sim q}\left[\log \frac{q(z)}{p(z)}\right]. \tag{6.5}$$

It is nonnegative and equals zero if and only if $q = p$. For the diagonal Gaussian encoder and isotropic Gaussian prior, the KL divergence has the closed form

$$\mathrm{KL}(q_\phi(z|y)\|p(z)) = \frac{1}{2}\sum_{j=1}^{\ell}\left(\mu_j^2 + \sigma_j^2 - 1 - \log\sigma_j^2\right), \tag{6.6}$$

where $\mu_j = [\mu_\phi(y)]_j$ and $\sigma_j = [\sigma_\phi(y)]_j$ are the j-th components of the encoder's mean and standard deviation outputs. This expression decomposes as a sum over latent dimensions because both distributions factorize across coordinates. We will exploit this decomposition property in Section 6.3.1 to combine different distribution families within a single latent space.

The reconstruction term in (6.4) encourages accurate reconstruction of observations, while the KL term regularizes the encoder by encouraging $q_\phi(z|y)$ to remain close to the prior. The weight β controls this trade-off: $\beta = 1$ gives the standard VAE, while $\beta \neq 1$ gives the β-VAE [160]. Without regularization ($\beta = 0$), the optimal encoder collapses to a Dirac delta at the point minimizing reconstruction error for each input, overfitting the training data rather than learning a smooth latent representation.

To enable gradient-based optimization of (6.4), the *reparametrization trick* [89, 138] expresses the latent sample as a deterministic function of the encoder parameters and independent noise:

$$z = \mu_\phi(y) + \sigma_\phi(y) \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \tag{6.7}$$

where $\odot$ denotes element-wise multiplication. Since ϵ is sampled independently of ϕ, the gradient $\nabla_\phi \mathbb{E}_{q_\phi}[\log p_\psi(y|z)]$ can be computed by backpropagation through the sampling operation. Section 6.3 generalizes this construction to non-Gaussian distributions.

6.2.3 The Topology Mismatch Problem

Two spaces are *homeomorphic* if there exists a continuous bijection with continuous inverse (a *homeomorphism*) between them, which implies they have the same topology. The standard Gaussian prior $p(z) = \mathcal{N}(0, I)$ has support $\mathcal{Z} = \mathbb{R}^\ell$, which is homeomorphic to Euclidean space. This is appropriate when the true latent space $\mathcal{M}$ is itself homeomorphic to $\mathbb{R}^n$, but creates a fundamental mismatch when $\mathcal{M}$ has a different topology. Many manifolds of practical interest are not homeomorphic to Euclidean space. These include bounded domains such as hypercubes $[0, 1]^n$, periodic domains such as tori $\mathbb{T}^n = (S^1)^n$, spheres S^{n-1} [139], and quotient spaces such as projective spaces and Möbius strips.

To see why the topology of $\mathcal{Z}$ matters, recall that both dimensionality reduction and generative modeling aim to capture the structure of $\mathcal{M}$. For dimensionality

reduction, we seek a mapping $g : f(\mathcal{M}) \to \mathcal{Z}$ such that $g \circ f : \mathcal{M} \to \mathcal{Z}$ is a homeomorphism. For such a homeomorphism to exist, $\mathcal{Z}$ must be homeomorphic to $\mathcal{M}$. This is the key requirement that motivates topology-aware VAEs. When $g \circ f$ is a homeomorphism but not the identity, the learned representation captures the manifold structure in a different coordinate system. For generative modeling, we often also desire *interpretability*: specific latent coordinates should correspond to meaningful factors of variation. This requires recovering the original coordinates $g(f(x)) = x$ for all $x \in \mathcal{M}$, which requires additional constraints (Section 6.3.4).

Before developing the solution, we address a natural question: *is explicit topology shaping even necessary?* Standard VAEs can already learn approximate nontrivial topologies through the reconstruction term alone. For example, a Gaussian VAE trained on data with a periodic latent factor may arrange its latent variables in a ring around the origin, approximating the circular topology. However, this arrangement is not stable under changes in the regularization strength: the KL term in (6.4) continuously pushes latent variables toward the prior's center, working against the data's periodic structure. Increasing the KL weight β eventually collapses the ring. Setting $\beta = 0$ avoids this collapse but removes all regularization, leading to overfitting and a latent space that does not generalize or support meaningful generation. In contrast, when the prior matches the data topology (e.g., a Wrapped Normal prior for a circular factor), the KL term is compatible with the manifold's structure, allowing strong regularization without topological distortion.

The topology mismatch problem can thus be stated as follows: *given a true latent space $\mathcal{M}$ with known topology, construct a VAE whose latent space $\mathcal{Z}$ is homeomorphic to $\mathcal{M}$, while retaining a tractable ELBO and end-to-end gradient-based training.*

6.3 Mathematical Framework for Topology Shaping

For the ELBO (6.4) to remain tractable when using non-Gaussian distributions, three conditions must be satisfied:

A) The KL divergence $\mathrm{KL}(q_\phi(z|y)\|p(z))$ must be computable, either in closed form or via efficient approximation. This requires $\mathrm{supp}(q_\phi) \subseteq \mathrm{supp}(p)$, since the KL divergence diverges when the encoder assigns positive probability to regions where $p(z) = 0$.

B) Samples $z \sim q_\phi$ must be expressible through the reparametrization trick as $z = T(\phi, \epsilon)$ with ϵ drawn from a fixed distribution independent of ϕ, so that the reconstruction term can be optimized via backpropagation.

C) All transformations in the computational graph must be differentiable, so that gradients propagate end-to-end.

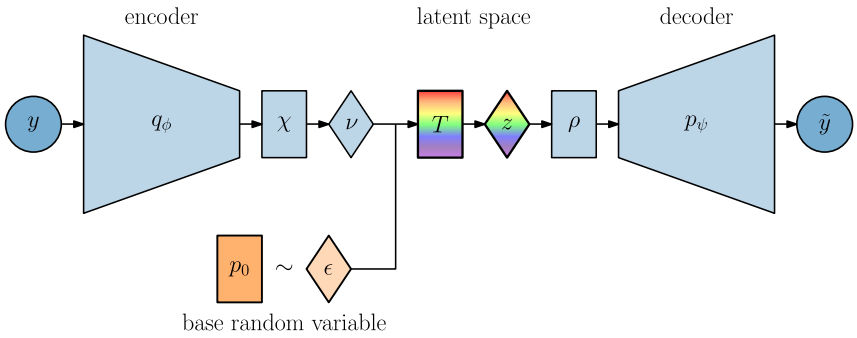


Figure 6.3. VAE architecture for topology shaping. The encoder maps observations y to distribution parameters ν, which define an encoder distribution $q_\phi(z|y)$ on target support $\mathcal{S}$ via coordinate transformation χ. The reparametrization T transforms base samples $\epsilon \sim p_0$ to latent samples z on the target support $\mathcal{S}$. The decoder receives z through feature map ρ, which embeds the latent coordinates in Euclidean space.

We now develop a framework that satisfies these conditions for all manifolds that admit a *product covering space*: manifolds that are either products of elementary factors, or quotients of such products by a finite symmetry group G. The four components of the framework (product topologies, distribution pairs, quotient constructions, and anchor constraints) are developed in Sections 6.3.1–6.3.4, with limitations discussed in Section 6.3.5. Figure 6.3 provides an overview of the generalized architecture.

6.3.1 Factorized Distributions and Product Topologies

The standard Gaussian VAE already exploits a factorized prior and encoder: both $p(z) = \mathcal{N}(0, I)$ and $q_\phi(z|y) = \mathcal{N}(\mu, \mathrm{diag}(\sigma^2))$ decompose across dimensions, and the KL divergence in (6.6) splits into a sum of per-dimension terms. This factorization is not specific to Gaussians. Whenever both encoder and prior factorize across latent coordinates, the KL divergence decomposes into independent per-factor terms, regardless of the distribution families used [159, Thm. 2.5.3]. This observation is the foundation of our framework: it allows us to shape the latent topology one coordinate at a time while keeping the KL tractable.

Proposition 6.1 (KL Divergence Decoupling)**.** *Let* $q(z) = \prod_{j=1}^{\ell} q_j(z_j)$ *and* $p(z) = \prod_{j=1}^{\ell} p_j(z_j)$ *be factorized distributions. Then*

$$KL(q\|p) = \sum_{j=1}^{\ell} KL(q_j\|p_j). \tag{6.8}$$

See Appendix 6.6.1 for the proof.

This decoupling has three important consequences. First, the KL divergence can be computed efficiently as a sum of per-factor divergences, each of which may have a different closed form depending on the distribution family. Second, different distribution families can be mixed across coordinates: one latent dimension can use a Gaussian while another uses a Wrapped Normal. Third, by selecting a distribution pair whose support matches the desired topology for each factor, we can shape the latent space factor by factor. This last point is formalized in the following proposition.

Proposition 6.2 (Product Topologies)**.** *Let* $q_\phi(z|y) = \prod_{j=1}^{\ell} q_j(z_j|y)$ *and* $p(z) = \prod_{j=1}^{\ell} p_j(z_j)$ *be factorized distributions where each pair* (q_j, p_j) *satisfies* $supp(q_j) \subseteq supp(p_j)$ *and has a computable KL divergence. Then the latent space has the product topology*

$$\mathcal{Z} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_\ell, \tag{6.9}$$

where $\mathcal{S}_j = supp(p_j)$ *is the support of the* j*-th coordinate's prior. The ELBO is tractable with the KL term decomposing as a sum over coordinates through Proposition 6.1.*

The support of a product distribution is the Cartesian product of the individual supports, which defines the product topology on $\mathcal{Z}$, and tractability follows from Proposition 6.1.

Although factorized distributions can only produce *product topologies*, the product construction is surprisingly versatile. Provided that reparametrizable distribution pairs with the right supports can be found, Propositions 6.1 and 6.2 immediately yield tractable VAEs on tori, cylinders, annuli, and arbitrary higher-dimensional products. Davidson et al. [140] applied the same principle to hyperspherical latent spaces, replacing a single high-dimensional vMF distribution with a product of lower-dimensional ones to sidestep the concentration bottleneck that arises at high dimension. The construction extends even further: many manifolds that are not themselves products (Möbius strips, Klein bottles, projective spaces) arise as quotients of products by discrete group actions. Constructing suitable distribution pairs for each support type and handling the identifications that quotient topologies require are nontrivial, however, and are the subjects of Sections 6.3.2 and 6.3.3.

One alternative is to drop the factorization assumption and estimate the KL divergence via Monte Carlo: $\mathrm{KL}(q\|p) = \mathbb{E}_{z\sim q}[\log q(z) - \log p(z)]$. Normalizing flows [161] provide the required reparametrization and tractable log-densities, but Monte Carlo estimation introduces gradient variance, and evaluating log-densities for expressive flows requires computing Jacobian determinants at each sample. We instead maintain factorized distributions with their closed-form KL divergences, and

handle non-product topologies through G-invariant decoder features in Section 6.3.3. The next section catalogs the specific distribution pairs available for each elementary factor type.

6.3.2 Reparametrizable Distribution Pairs

In this section, we catalog encoder-prior pairs that have computable KL divergences with matching supports, and that admit reparametrizable sampling (satisfying requirements A) and B) from Section 6.3). The key construction is the reparametrization trick [89]: expressing encoder samples as deterministic transformations of samples from a fixed base distribution. Let $\epsilon \sim p_0$ be a base random variable with support $\mathcal{S}_0$, and let $T : \mathcal{S}_0 \times \mathcal{V} \to \mathcal{S}$ be a smooth map parameterized by $\nu \in \mathcal{V}$. Then $z = T(\epsilon; \nu)$ defines a reparametrizable distribution on $\mathcal{S}$ with

$$\nabla_\nu \mathbb{E}_z[h(z)] = \mathbb{E}_{\epsilon \sim p_0}[\nabla_\nu h(T(\epsilon; \nu))], \tag{6.10}$$

for any differentiable function h, provided T is differentiable in ν. This moves the stochasticity into ϵ, allowing gradients to flow through T.

A classical construction takes $p_0 = \mathcal{U}(0, 1)$ and $T = F^{-1}$, where F is the CDF of the target distribution. This yields samples from any continuous distribution with invertible CDF. The Kumaraswamy and Exponential distributions in Table 6.1 use this approach. Other choices of p_0 and T are also valid: the Gaussian uses $p_0 = \mathcal{N}(0, 1)$ with affine $T(\epsilon; \mu, \sigma) = \mu + \sigma\epsilon$; the Wrapped Normal applies a wrapping operation to Gaussian samples; the von Mises-Fisher uses rejection sampling with reparametrized proposals [139].

Table 6.1 lists per-factor distribution pairs used in this chapter. Each entry specifies the support, the reparametrization T, and whether the KL divergence admits a closed form, an approximation, or requires Monte Carlo estimation. Combined with the factorized KL from Section 6.3.1, these pairs allow product topologies to be assembled factor by factor. This framework is not limited to one-dimensional factors: a factor can be a hypersphere S^{n-1} with a von Mises–Fisher distribution or a Euclidean space $\mathbb{R}^n$ with a multivariate Gaussian, as long as the distributions factorize; Table 6.1 focuses on low-dimensional building blocks because these suffice for our experiments. Closed-form KL expressions for each pair are provided in Appendix 6.6.

The table also includes the Gumbel-Softmax relaxation [162, 163], which extends the framework to discrete latent factors via approximate reparametrization of categorical variables, enabling hybrid discrete–continuous latent spaces [164]. All distribution pairs in Table 6.1 satisfy requirements A) and B). Requirement C) (smooth gradient flow) additionally demands that the encoder's real-valued outputs be mapped to valid parameter spaces without gradient pathologies, which is addressed next.

Table 6.1. Distribution Pairs for Topology Shaping

Encoder Distribution	**Support**	**Reparametrization** T	**Prior Distribution**	**KL**
Gaussian $\mathcal{N}(\mu, \sigma^2)$	$\mathbb{R}$	$\mu + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0,1)$	$\mathcal{N}(0,1)$	Closed
Exponential $\text{Exp}(\lambda_q)$	$[0, \infty)$	$-\log(1-u)/\lambda_q, \quad u \sim \mathcal{U}(0,1)$	$\text{Exp}(\lambda_0)$	Closed
Kumaraswamy $\text{Kum}(a, b)$	$[0, 1]$	$(1-(1-u)^{1/b})^{1/a}, \quad u \sim \mathcal{U}(0,1)$	$\mathcal{U}(0,1)$	Closed
Uniform $\mathcal{U}(a, b)$	$[a, b]$	$a + (b-a)u, \quad u \sim \mathcal{U}(0,1)$	$\mathcal{U}(a,b)$	Closed
Wrapped Normal $\text{WN}(\mu, \sigma^2)$	S^1	$\text{wrap}(\mu + \sigma\epsilon), \quad \epsilon \sim \mathcal{N}(0,1)$	$\mathcal{U}(S^1)$	Approximate
von Mises-Fisher $\text{vMF}(\mu, \kappa)$	S^{n-1}	Rejection sampling with Householder flow [139]	$\mathcal{U}(S^{n-1})$	Closed
Gumbel-Softmax $\text{GS}(\pi, t)$	$\{1, \ldots, K\}$	$\text{softmax}((\log \pi + g)/t)$, $g \sim \text{Gumbel}(0,1)$	$\text{Cat}(1/K)$	Closed

Encoder Coordinate Transformations

Neural networks produce outputs in $\mathbb{R}^n$, but the distribution parameters from Table 6.1 may live in non-Euclidean parameter spaces. We therefore need a coordinate transformation $\chi : \mathbb{R}^m \to \mathcal{V}$ that maps the encoder's real-valued outputs to valid parameters on the target support. This χ must cover all of $\mathcal{V}$ and be smooth with informative gradients everywhere in its domain. Naive approaches fail both requirements: clipping to a bounded range $[a, b]$ creates zero gradients at the boundaries, and taking outputs modulo 2π for periodic parameters creates gradient discontinuities at the wrap-around point.

For periodic parameters, we use a normalization-based construction. To output an angle $\mu \in S^1$, the encoder produces two real numbers $(c', s') \in \mathbb{R}^2$, which we normalize and convert:

$$(c, s) = \frac{(c', s')}{\|(c', s')\|}, \quad \mu = \arctan 2(s, c). \tag{6.11}$$

This defines $\chi : \mathbb{R}^2 \setminus \{0\} \to S^1$, which is smooth with well-defined gradients everywhere except the origin (a set of measure zero that is never reached in practice). For bounded parameters like the standard deviation $\sigma > 0$ of a Wrapped Normal, a softplus transformation $\chi(\cdot) = \log(1 + e^{(\cdot)})$ maps $\mathbb{R} \to \mathbb{R}_{>0}$ smoothly.

At this point, the framework appropriately transforms network outputs through the encoder and the variational sampling operation to produce latent samples on the target support while preserving differentiability. However, the decoder faces a different challenge: its input consists of the sampled latent coordinates themselves, which may have identifications that a standard neural network cannot respect. The next section addresses this.

6.3.3 Quotient Topologies via Invariant Features

After sampling z from the encoder distribution, we pass it to the decoder. In general, the latent coordinates may have *identifications*: distinct coordinate values that represent the same point. If $z_1 \sim z_2$ are identified, the decoder must satisfy $p_\psi(y \mid z_1) = p_\psi(y \mid z_2)$; otherwise, the model treats equivalent points inconsistently. Even the circle $S^1 = \mathbb{R}/2\pi\mathbb{Z}$ identifies $\theta \sim \theta + 2\pi$, while the Möbius strip also identifies $(h, \theta) \sim (1 - h, \theta + \pi)$. Our solution is to feed the decoder not the raw coordinates z, but a transformed representation using a *feature map* $\rho : \widetilde{\mathcal{Z}} \to \mathbb{R}^k$ that is invariant to the identifications.

We formalize identifications using group actions. Let $\widetilde{\mathcal{Z}}$ be a product space with a finite group G acting on it by isometries and freely, meaning that only the identity fixes any point, and let $\mathcal{Z} = \widetilde{\mathcal{Z}}/G$ be the quotient space where points in the same G-orbit are identified: $z_1 \sim z_2$ iff $z_2 = g \cdot z_1$ for some $g \in G$. The space $\widetilde{\mathcal{Z}}$ is called a covering space of $\mathcal{Z}$: informally, it is a "global unfolding" of $\mathcal{Z}$ in which identified

points are pulled apart. Note that while a given manifold may admit multiple covering spaces, we specifically require any covering space that is itself a product of elementary factors. This way, the factorized distributions from Section 6.3.1 apply on the encoder side. Each point in $\mathcal{Z}$ then has $|G|$ preimages in $\widetilde{\mathcal{Z}}$, so the encoder operates on the covering space while only the decoder input changes. For ρ to enforce the identifications, it must be (i) G-invariant, i.e., $\rho(z) = \rho(g \cdot z)$ for all $g \in G$; (ii) injective on G-orbits, so that distinct equivalence classes remain distinguishable; and (iii) smooth, so that gradients can flow through ρ during backpropagation. The following proposition guarantees that such a map always exists.

Proposition 6.3 (Quotient Topologies via Invariant Features)**.** *Let $\widetilde{\mathcal{Z}} \subseteq \mathbb{R}^m$ and let G be a finite group acting on $\widetilde{\mathcal{Z}}$ by isometries. Then there exists a smooth map $\rho : \widetilde{\mathcal{Z}} \to \mathbb{R}^k$ that is G-invariant and injective on G-orbits.*

Proof. For any two points z_1, z_2 in distinct G-orbits, the function $f(z) = \prod_{g \in G} \|z - g \cdot z_1\|^2$ is a polynomial in z (hence smooth), vanishes on the orbit of z_1, and is strictly positive on the orbit of z_2. It is G-invariant because G acts by isometries, so that $\|h \cdot z - hg \cdot z_1\| = \|z - g \cdot z_1\|$ and left multiplication by h only reorders the product. So G-invariant polynomials separate orbits. The ring of G-invariant polynomials is finitely generated by Hilbert's finiteness theorem [165, Thm. 2.1.3], so finitely many of them suffice to form an injective map ρ on the orbit space. As each component of ρ is a polynomial, ρ is smooth. □

Proposition 6.3 guarantees the existence of the required invariant features for any such G. For constructing ρ in practice, the *Reynolds operator* [165] is the central tool. Given any smooth function $f : \widetilde{\mathcal{Z}} \to \mathbb{R}$, its group average

$$R[f](z) = \frac{1}{|G|} \sum_{g \in G} f(g \cdot z) \tag{6.12}$$

is automatically G-invariant and smooth. Starting from a basis of smooth functions on the covering space (trigonometric for angular coordinates, polynomial for intervals), one applies R to monomials of increasing degree, discards dependent and vanishing terms, and verifies that the remaining features separate G-orbits. Noether's degree bound [165, Thm. 2.1.4] guarantees termination at degree at most $|G|$, which is degree two for the examples below. Appendix 6.7 gives the full procedure and worked examples. Since evaluating R requires only $|G|$ function evaluations, the construction cost is linear in the group order. Moreover, ρ is determined once at design time; during training it is a fixed closed-form function that adds no computational overhead beyond the increase in decoder input dimension. The following examples illustrate ρ for three cases.

Example 6.4 (Periodic coordinates)**.** *For a circular factor, the standard embedding* $\rho(\theta) = (\cos\theta, \sin\theta)$ *is the required invariant map. For product spaces without further identifications,* ρ *applies coordinate-wise: a cylinder* $S^1 \times [0,1]$ *uses* $\rho(\theta, h) = (\cos\theta, \sin\theta, h)$*. When the target manifold has additional identifications beyond periodicity, the feature maps compose accordingly.*

Example 6.5 (Möbius strip)**.** *The Möbius strip is* $([0,1]\times S^1)/\mathbb{Z}_2$ *with identification* $(h, \theta) \sim (1-h, \theta+\pi)$*. The shift* $\theta \mapsto \theta + \pi$ *has no fixed point on* S^1*, so the action is free. Applying the Reynolds operator to the cylinder decoder features* $\{\cos\theta, \sin\theta, h - \frac{1}{2}\}$ *(all of which flip sign under the deck transformation), one finds that degree-1 invariants vanish and the surviving degree-2 features are (see Appendix 6.7 for the full derivation):*

$$\rho(h,\theta) = \begin{pmatrix} \cos(2\theta) \\ \sin(2\theta) \\ (h-\frac{1}{2})^2 \\ \cos(\theta)(h-\frac{1}{2}) \\ \sin(\theta)(h-\frac{1}{2}) \end{pmatrix}. \tag{6.13}$$

From $\cos 2\theta$ *and* $\sin 2\theta$*, the angle* θ *is determined up to* θ *or* $\theta+\pi$*; the cross terms then resolve the remaining ambiguity, so* ρ *is injective on* $\mathbb{Z}_2$*-orbits.*

Example 6.6 (Klein bottle)**.** *The Klein bottle is* $\mathbb{T}^2/\mathbb{Z}_2$ *with identification* $(\theta_1, \theta_2) \sim (\theta_1 + \pi, -\theta_2)$*, which is free for the same reason. Applying the Reynolds operator to the torus decoder features* $\{\cos\theta_1, \sin\theta_1, \cos\theta_2, \sin\theta_2\}$ *yields the degree-2 invariant features:*

$$\rho(\theta_1,\theta_2) = \begin{pmatrix} \cos(2\theta_1) \\ \sin(2\theta_1) \\ \cos(\theta_2) \\ \cos(\theta_1)\sin(\theta_2) \\ \sin(\theta_1)\sin(\theta_2) \end{pmatrix}. \tag{6.14}$$

These features satisfy $\rho(\theta_1+\pi, -\theta_2) = \rho(\theta_1, \theta_2)$*, and the cross terms resolve the remaining ambiguity to ensure injectivity on* $\mathbb{Z}_2$*-orbits.*

Remark 6.7. *One might also consider the reverse construction: taking quotients factor-by-factor and then forming a product. This yields the same class of manifolds when each group acts on a single factor, since* $(\mathcal{S}_1/G_1)\times(\mathcal{S}_2/G_2) \cong (\mathcal{S}_1\times\mathcal{S}_2)/(G_1\times G_2)$*. The product-then-quotient formulation used here is strictly more general, as it also allows group actions that couple coordinates across factors (e.g., the* $\mathbb{Z}_2$ *action on the Möbius strip mixes* h *and* θ*).*

Sections 6.3.1–6.3.3 provide all the tools to build a latent space with a prescribed product or quotient topology. However, the learned coordinate system is typically

not unique: any self-homeomorphism of $\mathcal{Z}$ composed with the encoder is equally valid, and nothing in the training objective pins down which homeomorphism is learned.

6.3.4 Anchor Constraints

Anchor constraints resolve this coordinate ambiguity by specifying where particular observations should map in latent space. They can also create approximate topological holes, which is useful when the desired topology cannot be expressed as a product of bounded or periodic supports.

Reference Frame Anchoring

The coordinate ambiguity arises because the target latent space $\mathcal{Z}$ admits self-homeomorphisms: for example, $S^1 \times [0,1]$ is unchanged under any rotation of the angular coordinate, so the encoder and any such rotation of it are equally valid solutions. This issue is noted but left unresolved in prior work on non-Euclidean VAEs [141]. Anchor constraints fix the coordinate frame by penalizing deviation from prescribed locations. Writing $\mu_i = \mathbb{E}_{q_\phi}[z \mid y_i]$ for the posterior mean, the anchor loss is

$$\mathcal{L}_{\text{anchor}} = \frac{\lambda}{A} \sum_{i=1}^{A} \|\mu_i - z_i^*\|^2, \tag{6.15}$$

where $y_1, \ldots, y_A$ are anchor observations and $z_1^*, \ldots, z_A^*$ their prescribed target locations. For many distributions (including Gaussian and Kumaraswamy) the mean μ_i is available in closed form; otherwise Monte Carlo estimation can be used. The loss generalizes center loss [166], which attracts class features toward learnable centers; here the targets are fixed a priori for coordinate alignment.

More generally, anchoring need not cover the full latent vector. When $\mathcal{Z}$ decomposes as a product and only one factor carries the coordinate ambiguity (as with $S^1 \times [0,1]$, where the angular coordinate is rotationally ambiguous but the interval coordinate is not), the norm above need only be computed over the ambiguous coordinates. The remaining dimensions are left free to self-organize through the reconstruction objective. This reduces the number of labeled anchor observations required and avoids over-constraining coordinates with no ambiguity.

To break all symmetries, the anchors must be chosen such that no element of G other than the identity fixes all of them at once. For $S^1 \times [0,1]$ with $G = SO(2)$ acting by rotation of the circle, only the identity rotation fixes any given angle, so a single anchor suffices. If reflections are also admitted, so that $G = O(2)$, one anchor still leaves the reflection through it, and a second anchor that is neither equal nor antipodal to the first is needed.

Soft Topological Holes

Repulsive anchors can create approximate topological holes by penalizing proximity to specified locations. We seek a potential that diverges logarithmically at large distances (providing a long-range repulsive force) but remains finite at the origin (avoiding gradient singularities). A natural choice is the regularized log-potential

$$\mathcal{L}_{\text{repel}} = \frac{\lambda_r}{N} \sum_{i=1}^{N} \log\left(\frac{\|z_i - z^*\|^2}{\xi^2} + 1 \right), \tag{6.16}$$

where z_i are reparametrized samples from the encoder distribution, z^* is the center of the hole, and $\xi > 0$ controls the effective hole radius. The parameter ξ directly sets the crossover scale: points within distance ξ from z^* experience strong repulsion, while points beyond roughly 2ξ experience negligible effect. Unlike reference frame anchoring, which constrains the expected position, the repulsive loss operates on samples to ensure the entire posterior mass avoids the hole.

Unlike bounded distributions (Proposition 6.2), soft constraints cannot guarantee exact holes since the prior still has full support. They are useful when the desired topology cannot be expressed as a product or quotient of products.

6.3.5 Achievable Topologies and Limitations

Combining the tools from the preceding sections, the framework covers products (cylinders, tori, hypercubes), quotients of products (Möbius strips, Klein bottles, projective spaces), and single-factor manifolds such as hyperspheres S^{n-1} (via the von Mises-Fisher distribution). Because $SO(3) \cong \mathbb{RP}^3 = S^3/\mathbb{Z}_2$, even rotational latent spaces fall within the achievable class. Discrete factors can also be included via the Gumbel-Softmax relaxation (Section 6.3.2). Several limitations remain, however. The topology must be specified a priori; we do not learn it from data. Disconnected manifolds require mixture priors [167], breaking KL decoupling and requiring techniques such as importance-weighted bounds [168]. Finally, manifolds whose universal covering space is not a product of our elementary factors (such as higher-genus surfaces and hyperbolic manifolds) cannot be represented exactly. Nevertheless, many manifolds of practical interest fall within the achievable classes.

The next section empirically validates the framework on several such topologies.

6.4 Experiments

We validate the framework in two parts. Section 6.4.2 presents synthetic experiments on cylinders, tori, and Möbius strips, where the true latent space is known, so topological fidelity can be assessed quantitatively. Sections 6.4.3 and 6.4.4 demonstrate the framework on manipulations of MNIST images, specifically rotation and

translation, which both induce periodic latent coordinates. Before showing results, we describe the evaluation metrics in Section 6.4.1.

Because different latent topologies interact differently with the KL regularization, a fixed β does not yield a fair comparison. We therefore tune β separately for each model so that the *unweighted* KL divergence $\mathrm{KL}(q_\phi \| p)$ is approximately equal across all models within each experiment. This ensures that the models operate at comparable regularization strength and that differences in reconstruction or topology metrics are not artifacts of one model being less regularized than another. The resulting β values and unweighted KL terms are reported in each experiment's table. All experiments use an 80/20 train/test split, and each topology-aware model is compared to a Gaussian VAE baseline with the same total latent dimensionality ℓ.

The VAE training objective is typically nonconvex, so we repeat each synthetic configuration over 30 random initializations and report medians and interquartile ranges. The dataset and the train/test split are held fixed, so the spread reflects optimization alone, and each synthetic figure shows the run closest to the across-seed median. The image experiments are too expensive to repeat and are reported as single runs.

Within an experiment, every model shares the same encoder and decoder architecture, the same training schedule, and the same data. Only the latent distribution and the decoder feature map ρ differ, so the comparison isolates the effect of the prescribed topology. The synthetic models use multilayer perceptrons with encoder widths $[50, 128, 128, 64]$ and mirrored decoders, trained for 8000 epochs with Adam at a learning rate of 10^{-3}. The image models use a convolutional backbone with hidden width 512 for rotated MNIST and 256 for shifted MNIST, trained for 50 epochs with Adam at learning rates 3×10^{-3} and 10^{-3} respectively, with a batch size of 256. In every run β is annealed linearly from zero over the first quarter of training. The anchor weight λ is 10 on the synthetic manifolds, 50 for rotated MNIST, and 100 for shifted MNIST.

6.4.1 Evaluation Metrics

The tables in each experiment report four quantities alongside the tuned β. The first two, train and test RMS reconstruction error, measure how well the VAE reconstructs observations. The third, the unweighted KL divergence, quantifies regularization strength. These three quantities are standard but do not directly assess whether the learned latent space respects the assumed topology, since a model with low reconstruction error might still scramble the latent geometry.

The fourth quantity tests whether pairwise distances between latent variables reflect the true distances on the data manifold. Since every experiment provides the true latent variables x_i, we can compute geodesic distances on $\mathcal{M}$ and compare

them to geodesic distances between latent variables. A natural measure for this comparison is Kruskal's stress-1 [169, p. 8], which is standard in multidimensional scaling, here normalized by the true distances:

$$\sigma_{\text{geo}} = \sqrt{\frac{\sum_{i\neq j}(\bar{d}_{ij} - \bar{\delta}_{ij})^2}{\sum_{i\neq j}\bar{d}_{ij}^2}}, \tag{6.17}$$

where d_{ij} is the geodesic distance between true latent variables x_i, x_j on $\mathcal{M}$ and δ_{ij} is the geodesic distance between the posterior means z_i, z_j on the model's latent space. Bars denote mean-normalization: $\bar{d} = d/\mathbb{E}[d]$, which removes scale differences. We call this the *geodesic stress*. For the cylinder, the latent geodesic uses the product metric on $S^1 \times [0,1]$; for the torus, the product of two angular distances. For quotient manifolds, the geodesic distance is the minimum over the group orbit: $d_{\mathcal{M}}(z_1, z_2) = \min_{g\in G} d_{\widetilde{\mathcal{Z}}}(z_1, g\cdot z_2)$. For the Möbius strip with $G = \mathbb{Z}_2$, this amounts to computing two covering-space distances and taking the smaller one. In the image experiments the known true factor is the applied rotation or shift, so the comparison is restricted to the angular factor for every model: the topology-aware models are read on their circular coordinates, and the Gaussian on whichever coordinate plane gives it the lowest stress. Lower stress indicates better preservation of the manifold's distance structure.

6.4.2 Synthetic Validation: Cylinder, Torus, and Möbius Strip

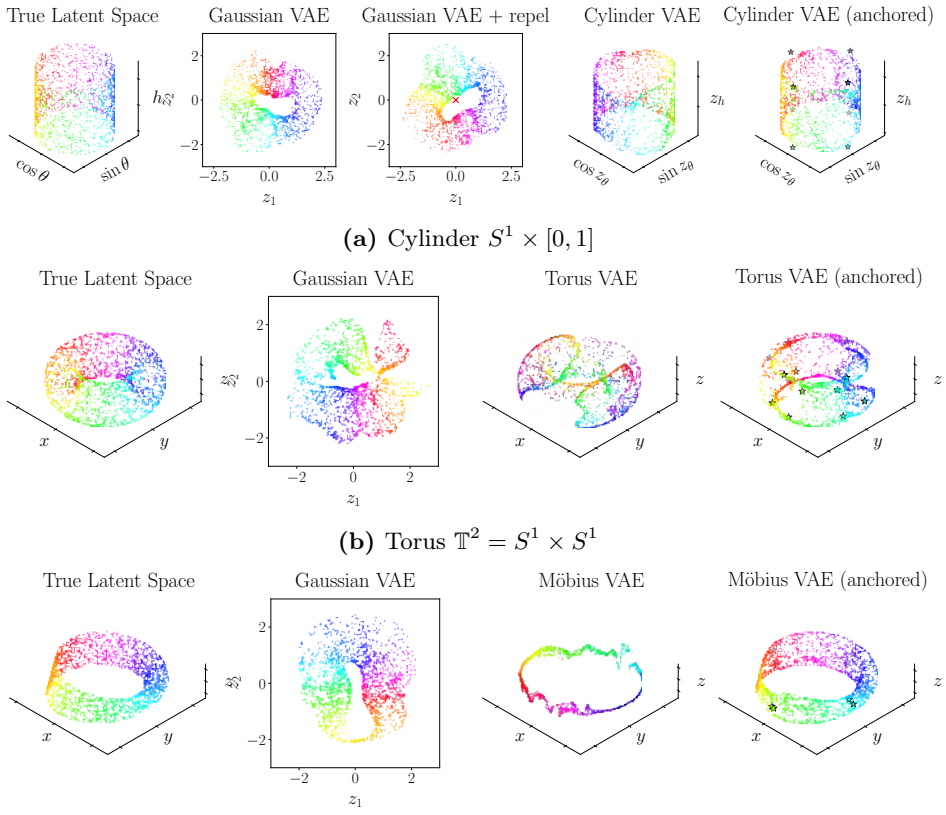
For each of the three target manifolds, we draw $N = 2000$ true latent variables $x \in \mathcal{M}$ uniformly and map each to an observation $y \in \mathbb{R}^{50}$ through a fixed, nonlinear and injective generative network $f_\star$ (three layers with tanh and LeakyReLU nonlinearities, fixed weights) with additive Gaussian noise of standard deviation 0.1, as in (6.1). The network is shared across all three experiments, only the input coordinates differ. Models are trained with β tuned so that the unweighted KL is approximately equal within each experiment. Table 6.2 collects reconstruction and topology metrics across all three experiments. Figure 6.4 shows the corresponding learned latent representations.

Cylinder

The cylinder $S^1 \times [0,1]$ is the simplest mixed-product example: one periodic coordinate and one bounded coordinate. We draw $\theta \sim \mathcal{U}(-\pi, \pi)$ and $h \sim \mathcal{U}(0,1)$, embed the pair in $\mathbb{R}^3$ as $(\cos\theta, \sin\theta, h)$, and apply $f_\star$. Four models are compared: a standard Gaussian VAE; a Gaussian VAE augmented with a repelling anchor at the origin (soft topological hole); an unanchored Cylinder VAE with a Wrapped

Table 6.2. Synthetic experiments: reconstruction and topology metrics. Lower is better for the RMS and stress metrics, and the best value within each topology is shown in bold for those columns. Each entry is the median over 30 seeds, with the interquartile range in brackets for test RMS and geodesic stress. The weight β is tuned per model so that the unweighted KL divergence is matched within each topology, which the KL column reports.

Topology	Model	β	Train RMS	Test RMS	KL	Geodesic Stress
Cylinder $S^1 \times [0, 1]$	Gaussian	1.20	**0.118**	0.188 [0.187, 0.191]	3.20	0.316 [0.315, 0.324]
	Gaussian + repel	1.20	**0.118**	0.189 [0.185, 0.193]	3.19	0.343 [0.339, 0.349]
	Cylinder	1.00	0.121	**0.181** [0.179, 0.189]	3.23	0.081 [0.079, 0.121]
	Cylinder (anch.)	1.00	0.121	**0.181** [0.179, 0.183]	3.23	**0.074** [0.073, 0.075]
Torus $\mathbb{T}^2$	Gaussian	1.00	0.139	0.206 [0.200, 0.208]	4.05	0.409 [0.388, 0.417]
	Torus	1.00	0.136	**0.198** [0.195, 0.201]	4.12	0.292 [0.268, 0.321]
	Torus (anch.)	1.00	**0.135**	**0.198** [0.195, 0.200]	4.13	**0.242** [0.240, 0.246]
Möbius strip	Gaussian	1.30	**0.091**	0.179 [0.175, 0.183]	3.33	0.332 [0.326, 0.341]
	Möbius	0.95	0.153	0.205 [0.201, 0.208]	3.25	0.235 [0.233, 0.237]
	Möbius (anch.)	0.95	0.092	**0.165** [0.163, 0.166]	3.46	**0.087** [0.086, 0.088]



(a) Cylinder $S^1 \times [0, 1]$

(b) Torus $\mathbb{T}^2 = S^1 \times S^1$

(c) Möbius strip

Figure 6.4. Learned latent representations for the three synthetic experiments, each from the representative run of Table 6.2. Every panel shows, from left to right, the true factors, the Gaussian baseline, the topology-aware VAE, and the topology-aware VAE with anchoring; the cylinder in **(a)** carries an additional column for the repelling-anchor Gaussian. Anchor points are marked $\star$. Three-dimensional panels show embeddings in ambient $\mathbb{R}^3$ space, and the color of the points corresponds to the first true latent coordinate. For the cylinder in **(a)**, the embedding is given by $(\cos\theta, \sin\theta, h)$ or $(\cos z_\theta, \sin z_\theta, z_h)$. For the torus in **(b)**, the embeddings use standard parametrizations with ambient coordinate axes. The Möbius strip in **(c)** first maps each point to its canonical representative in the fundamental domain $[0, 1] \times [-\pi, \pi)$, then applies the standard embedding $(\cos\theta, \sin\theta, h)$ to place it on the strip surface. The collapse of the height coordinate discussed in Section 6.4.2 is visible as the flattening of the unanchored Möbius panel.

Normal encoder for θ and a Kumaraswamy encoder for h; and an anchored version with eight anchors on the grid $\theta \in \{-\pi, -\frac{\pi}{2}, 0, \frac{\pi}{2}\}$, $h \in \{0, 1\}$.

Figure 6.4**(a)** shows the learned representations. The Gaussian VAE cannot represent the periodic and bounded structure of the cylinder: the angular coordinate wraps without constraint and the height is unbounded in the latent space, so points that coincide at $\theta = \pm\pi$ are assigned different latent positions. The Cylinder VAE correctly captures both structural features of the target: the height coordinate is confined to $[0, 1]$ by the Kumaraswamy support and the angular coordinate is periodic. Without anchoring, however, the coordinate system is arbitrary and the learned angular frame may be rotated relative to the true one; adding anchor constraints aligns the learned frame with the true latent variables.

At matched KL, the Gaussian baseline fits the training data marginally better than either Cylinder VAE, and yet both Cylinder VAEs reconstruct held-out data more accurately. The cylindrical latent space therefore generalizes better despite the slightly worse fit, which is what a prior matched to the data manifold is meant to provide. Its geodesic stress is lower by roughly a factor of four. The cylinder is a product manifold, so the framework prescribes its topology directly and the repelling anchor is not the tool intended for it. It improves neither error, so approximating a hole does not substitute for prescribing the topology.

The spread across seeds separates the two Cylinder VAEs. Anchored, every seed reaches a geodesic stress within a narrow band. Unanchored, 7 of the 30 seeds score as poorly as the Gaussian: those runs recover the periodic structure but settle into a folded angular frame, which the anchor constraints preclude.

Torus

The torus $\mathbb{T}^2 = S^1 \times S^1$ is a purely periodic product: both latent coordinates are angles. We draw $\theta_1, \theta_2 \sim \mathcal{U}(-\pi, \pi)$, embed the pair on a torus surface in $\mathbb{R}^3$ with major radius 2 and minor radius 0.8 via $((2 + 0.8\cos\theta_2)\cos\theta_1,\ (2 + 0.8\cos\theta_2)\sin\theta_1,\ 0.8\sin\theta_2)$, and apply $f_\star$. Three models are compared: a Gaussian VAE; a Torus VAE with Wrapped Normal encoders for both angular coordinates; and an anchored Torus VAE with 16 anchor points on a 4×4 grid over $[-\pi, \pi]^2$.

Figure 6.4**(b)** shows the learned representations. The Gaussian VAE produces boundary artifacts at $\theta_1, \theta_2 \approx \pm\pi$: points that are close on the torus surface but near opposite ends of the angular range are placed far apart in the Gaussian latent space, because the Gaussian prior does not wrap either dimension. The Torus VAE distributes points smoothly across the full $[-\pi, \pi]^2$ range.

At matched KL, both Torus VAEs reconstruct better than the Gaussian baseline and lower the geodesic stress substantially, with anchoring giving a further gain. The uniform prior on $\mathbb{T}^2$ is matched by any encoder that fills the torus, whatever angular frame that encoder chooses, so the prior alone leaves the frame arbitrary.

Anchoring pins it down, and the further drop in geodesic stress is what registers the difference.

Möbius Strip

The Möbius strip is homeomorphic to $([0,1] \times S^1)/\mathbb{Z}_2$ under the identification $(h, \theta) \sim (1-h, \theta+\pi)$. We generate observations by sampling $\theta \sim \mathcal{U}(-\pi, \pi)$ and $h \sim \mathcal{U}(0,1)$ on the covering cylinder, computing the $\mathbb{Z}_2$-invariant features $\rho(h,\theta)$ from (6.13), and applying $f_\star$ to those features. Because $\rho(h,\theta) = \rho(1-h, \theta+\pi)$, the two covering-space preimages of each Möbius point produce the same observation up to noise, which is the correct generative semantics for the quotient identification.

The Möbius VAE encodes into the covering cylinder $S^1 \times [0,1]$ and passes the $\mathbb{Z}_2$-invariant features to the decoder. Because each Möbius point has two preimages, the KL divergence requires a correction: writing $r = q_C(\tau(h,\theta))/q_C(h,\theta)$ for the density ratio at identified points (where $\tau(h,\theta) = (1-h, \theta+\pi)$ is the deck transformation), the adjusted Möbius KL is

$$\mathrm{KL}(q_M \| p_M) = \mathrm{KL}(q_C \| p_C) - \log 2 + \mathbb{E}[\log(1+r)]. \tag{6.18}$$

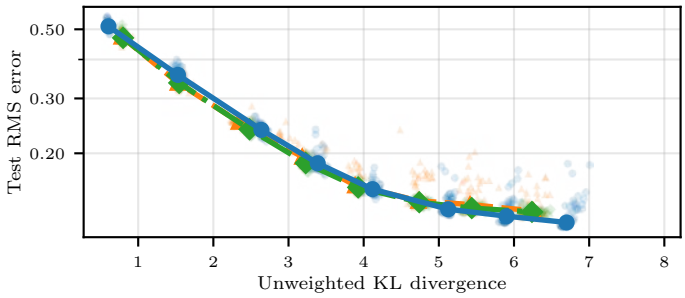
See Appendix 6.6.7 for the derivation.

Figure 6.4**(c)** shows the learned representations. The Gaussian VAE cannot capture the non-orientable structure of the Möbius strip at all. The Möbius VAE without anchoring correctly uses the Möbius topology in principle. In practice it collapses the height coordinate toward $h \approx 0.5$ to reduce the KL term, and it does so in all 30 seeds. The collapsed model reconstructs worse than the Gaussian baseline on both train and test data, so the matched topology on its own is of no use here.

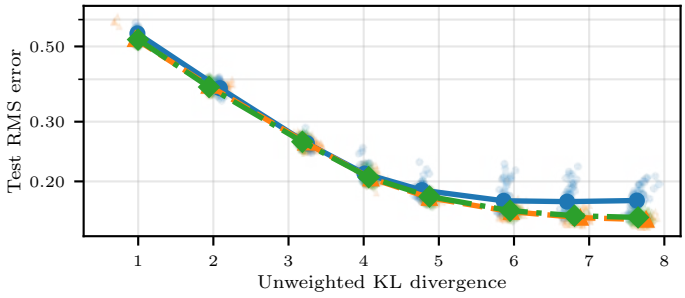
Anchoring removes the collapse in all 30 seeds. The anchored model recovers Gaussian-level train error while retaining the full Möbius structure, and it yields both the best test error and the lowest geodesic stress of the three. The anchor penalty is a non-negative addition to the objective, so it cannot lower the global optimum. Its benefit is therefore a statement about optimization: the anchors steer training away from the degenerate solution that otherwise traps the model. Here the prescribed topology delivers nothing without the anchors that make it reachable.

KL Sweep Analysis

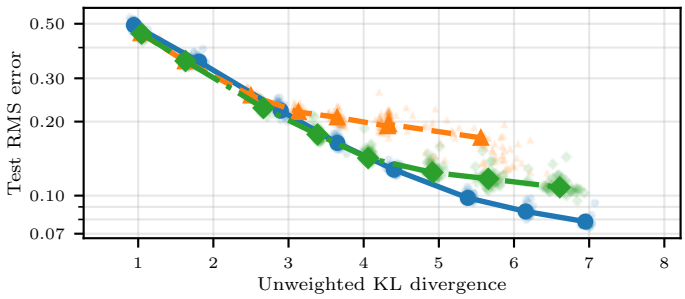
The tables above compare the models at one operating point. To assess whether the topology advantage persists across the full reconstruction/regularization trade-off [170], we vary the KL weight β from 0.05 to 10.0 over eight values and train every model at each setting over 50 random seeds per topology. Figure 6.5 plots test reconstruction RMS against unweighted KL divergence, so that models are



(a) Cylinder $S^1 \times [0, 1]$



(b) Torus $\mathbb{T}^2$



(c) Möbius strip

Figure 6.5. KL sweep for the three synthetic topologies: the cylinder $S^1 \times [0, 1]$ in **(a)**, the torus $\mathbb{T}^2$ in **(b)**, and the Möbius strip in **(c)**. In each, β varies from 0.05 to 10.0 and test reconstruction RMS is plotted against unweighted KL divergence for the Gaussian baseline (—●—), the topology-aware VAE (- -▲- -), and the anchored topology-aware VAE (- -◆- -). Lines connect medians over 50 seeds per β; the faint points behind them are the individual seeds. Both axes decrease toward the lower left, so a curve lying below and to the left is better. A smaller β yields a larger KL divergence, which places the weakly regularized settings on the right of each panel.

compared at equal regularization rather than at equal β. The sweep trains 3600 models in total, so it uses a reduced setting relative to Table 6.2: 800 samples instead of 2000, half the hidden width, and a single shared β per point rather than a per-model β tuned for matched KL. Absolute errors are therefore slightly higher than in the table, and the comparison of interest is between the curves within a panel.

The Torus VAE reconstructs better than the Gaussian baseline at every KL value in the sweep. For the cylinder and the Möbius strip the curves cross, at a KL of roughly 4.8 and 4.1 respectively: the anchored topology-aware model is ahead below those values, the Gaussian above them, and the margin widens as regularization strengthens. On the Möbius strip the unanchored model stays behind over most of the sweep, for the reason given in Section 6.4.2. The region in which the Gaussian wins is the one where the KL term barely constrains the encoder, so the prior has little influence on the representation and correspondingly little opportunity to help. The operating points of Table 6.2 lie on the favorable side of both crossings.

The synthetic experiments validated the framework on low-dimensional observations generated from known manifolds. We now turn to real images to assess whether the topology benefits transfer to a higher-dimensional, more complicated setting.

6.4.3 Rotated MNIST ($\mathbb{R}^2 \times S^1$)

In this experiment, each image is a 28×28 MNIST digit rotated by a random angle $\theta \sim \mathcal{U}(-\pi, \pi)$. The two generative factors are the digit class and the rotation angle, a single S^1 factor. We use digit classes 4 and 7, which are both asymmetric under rotation so that the S^1 coordinate is identifiable, with 500 exemplars per class, each rotated 180 times, giving $N = 180\,000$ images with $y_i \in \mathbb{R}^{784}$.

All three models have the same total latent dimensionality $\ell = 3$. For the Mixed-Circle VAE, the learned latent variable is $z = (z_1, z_2, z_\theta)$, where $z_\theta \in S^1$ is the circular coordinate and $(z_1, z_2) \in \mathbb{R}^2$ are the Euclidean style coordinates; for the Gaussian, all three coordinates are Euclidean. The three models are:

1. Gaussian VAE with $\ell = 3$, which yields a latent topology of $\mathbb{R}^3$
2. Unanchored Mixed-Circle VAE, which yields a latent topology of $\mathbb{R}^2 \times S^1$;
3. Anchored Mixed-Circle VAE, which yields a latent topology of $\mathbb{R}^2 \times S^1$.

The Mixed-Circle VAE uses a Wrapped Normal encoder for z_θ and a standard Gaussian encoder for (z_1, z_2). The anchors are applied only on the circular coordinate, see Section 6.3.4, with 90 evenly spaced target angles per digit class.

Table 6.3. Rotated MNIST Experiment: Reconstruction and Topology Metrics. Lower is better for the RMS and stress metrics, and the best value in each of those columns is shown in bold. Unlike Table 6.2, each row is a single training run, so differences of a few percent are not resolved.

Model	β	Train RMS	Test RMS	KL	Geodesic Stress
Gaussian ($\ell = 3$)	0.02	0.134	0.140	16.52	0.675
Mixed-Circle VAE	0.02	0.143	0.146	15.22	0.682
Mixed-Circle VAE (anch.)	0.02	**0.126**	**0.131**	15.16	**0.253**

The coordinates (z_1, z_2) are unconstrained and organize by digit class through reconstruction pressure alone.

Figure 6.6 compares the three models in four rows. Row 1 confirms that the rotation angle is encoded. The Mixed-Circle VAE assigns it to the native S^1 coordinate z_θ, while the Gaussian distributes it across all three Euclidean dimensions. Only the anchored version retains the orientation of the true data. Row 2 shows the style coordinates, in which the digit classes occupy distinguishable regions for every model. Row 3 shows a 12-step decoded ring per digit class, with the style coordinates (z_1, z_2) fixed at the class centroid and z_θ swept uniformly over S^1. Fixing the centroid isolates the rotation factor and restricts the displayed digits to a single class. The Mixed-Circle VAE produces coherent digits at every angle and wraps back cleanly. Row 4 shows decoded geodesics, obtained by interpolating between two endpoints of the same digit class and decoding at each step. The Mixed-Circle VAE stays in distribution along the whole path. In both rows the Gaussian leaves the data distribution, producing blurred or implausible digits. Rows 3 and 4 are generative rather than reconstructive, since every image shown is decoded from a latent position that no observation occupies. Sweeping z_θ over the circle therefore yields valid digits at any rotation, which the Gaussian latent space does not support.

Table 6.3 reports reconstruction and topology metrics. The anchored Mixed-Circle VAE reconstructs more accurately than the Gaussian and lowers the geodesic stress by nearly two thirds, and it does so while operating at a lower KL than the baseline. The advantage of a matched topology therefore carries over from the synthetic manifolds to real images. The unanchored model leaves the geodesic structure almost untouched, which places the gain with the anchors rather than with the circular coordinate alone.

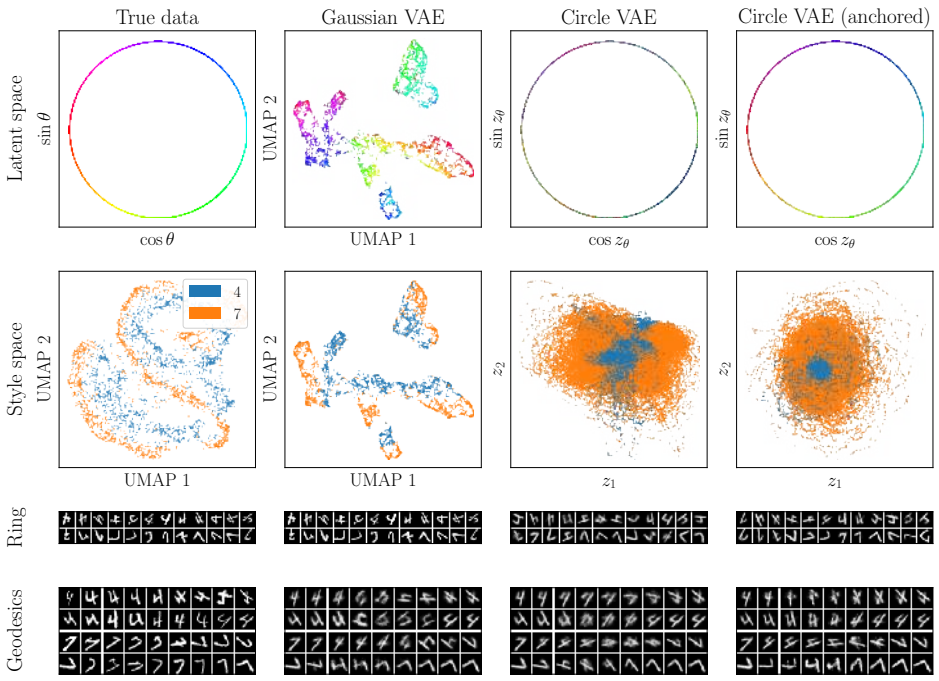


Figure 6.6. Rotated MNIST experiment ($\mathbb{R}^2 \times S^1$ latent space, digit classes 4 and 7). Columns: ground truth, Gaussian VAE ($\ell = 3$), Mixed-Circle VAE, and Mixed-Circle VAE (anchored). First row: latent coordinates colored by the true rotation angle θ. For the Mixed-Circle VAEs, the circular coordinates $(\cos z_\theta, \sin z_\theta)$ are plotted; for the Gaussian, a UMAP projection of the full latent space is shown instead; for the ground-truth column, $(\cos\theta, \sin\theta)$ is plotted. Second row: latent coordinates colored by digit class. The Mixed-Circle VAE shows the Euclidean coordinates (z_1, z_2), while a UMAP projection is shown for the ground-truth and Gaussian columns. Third row: 12-step ring for each digit class (4 and 7), sweeping z_θ for the Mixed-Circle VAEs or the proxy $\mathrm{atan2}(z_2, z_1)$ for the Gaussian, with style coordinates (z_1, z_2) fixed at the class centroid. Fourth row: decoded geodesics between two endpoints from the same digit class, linearly interpolated in latent space.

6.4.4 Shifted MNIST ($\mathbb{R}^2 \times \mathbb{T}^2$)

In this experiment, images are generated by cyclically shifting MNIST digits from classes 3, 4, and 7 by random integer pixel offsets (d_x, d_y) with $d_x, d_y \in \{0, \ldots, 27\}$ using `numpy.roll`. The true latent coordinates are $\theta_i = (2\pi d_i/28 + \pi) \bmod 2\pi - \pi \in [-\pi, \pi)$ for $i \in \{1, 2\}$, placing the pair (θ_1, θ_2) exactly on $\mathbb{T}^2 = S^1 \times S^1$. We use 500

exemplars per digit class with 100 random shifts each, giving $N = 150\,000$ images.

All three models have the same total latent dimensionality $\ell = 4$. The Mixed-Torus VAE has learned latent variable $z = (z_1, z_2, z_{\theta_1}, z_{\theta_2})$, with Wrapped Normal encoders on the circle coordinates $z_{\theta_1}, z_{\theta_2} \in S^1$ and a standard Gaussian encoder on the style coordinates $(z_1, z_2) \in \mathbb{R}^2$. The Gaussian VAE uses a flat $\mathbb{R}^4$ prior.

As in the rotated MNIST experiment (Section 6.4.3), only the angular coordinates z_{θ_1} and z_{θ_2} are anchored, while the style coordinates (z_1, z_2) are unconstrained. Anchor targets correspond to pixel shifts $(d_x, d_y) \in \{0, 7, 14, 21\}^2$, giving 16 exact angular targets per digit class.

Figure 6.7 compares the three models in four rows. Row 1 confirms that both shift angles are encoded. The Mixed-Torus VAEs assign them to native S^1 coordinates, while the Gaussian distributes them across all four Euclidean dimensions. The anchored model recovers the true parameter space almost exactly, which makes its latent coordinates directly interpretable. Row 2 shows the style coordinates, in which the three digit classes occupy distinguishable regions. Row 3 shows a 10×10 decoded grid, using the same class-centroid approach as in Section 6.4.3. Here the anchored Mixed-Torus VAE reproduces the ground-truth shift pattern, while the Gaussian shows no periodic structure at all. Row 4 shows decoded geodesics for each digit class, constructed as in Section 6.4.3. The Mixed-Torus VAE again interpolates smoothly along the whole path. As in the rotated experiment, these two rows are generated from latent positions rather than reconstructed from data, so the grid in row 3 amounts to sampling the full $\mathbb{T}^2$ of shifts for a fixed digit style.

Table 6.4 reports reconstruction and topology metrics. Reconstruction quality is similar across all three models, with the unanchored Mixed-Torus VAE marginally ahead while operating at a 10% higher KL than the other two. The anchored Mixed-Torus VAE attains by far the lowest geodesic stress, and therefore best reproduces the periodic structure this experiment is built around. Without anchoring, the geodesic stress rises above the Gaussian baseline: the arbitrary angular frame of Section 6.4.2 reappears on real images.

6.5 Discussion and Conclusions

We have presented a framework for designing VAE latent spaces with prescribed non-Euclidean topologies. The design procedure has four steps: decompose a target manifold into elementary factors, assign a compatible encoder-prior distribution pair to each, construct a feature map ρ that embeds latent coordinates in Euclidean space for the decoder, and add anchor constraints when coordinate alignment is needed. Factorized distributions yield product topologies with closed-form KL divergences, and G-invariant feature maps handle quotient identifications (Proposition 6.3). All transformations are differentiable, preserving gradient flow throughout.

Table 6.4. Shifted MNIST Experiment: Reconstruction and Topology Metrics. Lower is better for the RMS and stress metrics, and the best value in each of those columns is shown in bold. As in Table 6.3, each row is a single training run.

Model	β	Train RMS	Test RMS	KL	Geodesic Stress
Gaussian ($\ell = 4$)	0.04	0.174	0.182	19.29	0.240
Mixed-Torus VAE	0.02	**0.165**	**0.175**	21.15	0.351
Mixed-Torus VAE (anch.)	0.02	**0.165**	0.179	19.37	**0.149**

Prescribing an anchored topology improves the geodesic stress in all five experiments, by between 38% and 77% against a Gaussian baseline at comparable KL divergence. The anchored topology-aware model also reconstructs held-out data more accurately than the Gaussian in all five, by up to 8%. On the cylinder it does so while fitting the training data marginally worse, so the gain is one of generalization rather than of capacity. The KL sweep locates the regime in which the reconstruction gain holds. Once β exceeds a small topology-dependent threshold, the anchored topology-aware models reconstruct better. Below it, the KL term barely constrains the encoder, so the prior has little opportunity to help. The decoded sweeps in Figures 6.6 and 6.7 point to the same advantage for generation. Every image in those rows is decoded from a latent position that no observation occupies, and the topology-aware models produce plausible digits across the whole manifold while the Gaussian degrades away from the data.

Two lessons can be taken from the experiments. The first is that a matched prior fixes the topology of the latent space but not the coordinate frame within it. A uniform prior on the circle or the torus is matched by any encoder that fills it, whatever rotation that encoder settles on, so the training objective alone leaves the frame free. Geodesic stress is what exposes this, since it compares against the true coordinates rather than against the prior. Anchor constraints are therefore part of the pipeline rather than an optional refinement.

The second is that the anchor constraints also decide which representation training reaches at all. Without them, roughly a quarter of the cylinder seeds settle into a folded angular frame, and the Möbius model discards its height coordinate in every run. Since the penalty cannot lower the global optimum (Section 6.4.2), anchoring is worth applying whenever the target manifold admits a symmetry.

Several directions for future work remain. The topology must currently be specified a priori. Learning it from data instead is an open problem. The invertible transformation perspective connects naturally to normalizing flows [148], where the

transformation is learned rather than fixed. The quotient topology approach could be extended to continuous symmetry groups or to learning the group action from data. Combining topology shaping with disentanglement objectives may enable latent spaces that are both topologically faithful and interpretable.

6.6 Appendix: KL Divergence Derivations

This appendix provides the proof of Proposition 6.1, closed-form KL divergences for the distribution pairs in Table 6.1, and the derivation of the Möbius strip KL divergence, which requires special treatment due to the quotient topology.

6.6.1 KL Divergence Decoupling (Proof of Proposition 6.1)

By definition of KL divergence and the factorization assumption:

$$\begin{aligned}
\mathrm{KL}(q\|p) &= \int \prod_{j=1}^{\ell} q_j(z_j) \log \frac{\prod_{j=1}^{\ell} q_j(z_j)}{\prod_{j=1}^{\ell} p_j(z_j)}\, dz \\
&= \int \prod_{j=1}^{\ell} q_j(z_j) \sum_{k=1}^{\ell} \log \frac{q_k(z_k)}{p_k(z_k)}\, dz \\
&= \sum_{k=1}^{\ell} \int q_k(z_k) \log \frac{q_k(z_k)}{p_k(z_k)} \prod_{j \neq k} q_j(z_j)\, dz \\
&= \sum_{k=1}^{\ell} \mathrm{KL}(q_k\|p_k), \qquad (6.19)
\end{aligned}$$

where the last step uses $\int q_j(z_j)\, dz_j = 1$ for all $j \neq k$.

6.6.2 Gaussian-Gaussian KL

The KL divergence between univariate Gaussians $q = \mathcal{N}(\mu_q, \sigma_q^2)$ and $p = \mathcal{N}(\mu_p, \sigma_p^2)$ is:

$$\mathrm{KL}(q\|p) = \log \frac{\sigma_p}{\sigma_q} + \frac{\sigma_q^2 + (\mu_q - \mu_p)^2}{2\sigma_p^2} - \frac{1}{2}. \qquad (6.20)$$

For the standard prior $p = \mathcal{N}(0, 1)$ used in vanilla VAEs, this simplifies to $\frac{1}{2}(\mu_q^2 + \sigma_q^2 - 1 - \log \sigma_q^2)$.

6.6.3 Exponential-Exponential KL

For the half-infinite latent space $\mathbb{R}_{\geq 0}$, we use the Exponential distribution. Given $q = \text{Exp}(\lambda_q)$ and $p = \text{Exp}(\lambda_p)$ with rate parameters $\lambda_q, \lambda_p > 0$:

$$\text{KL}(q\|p) = \log \frac{\lambda_q}{\lambda_p} + \frac{\lambda_p}{\lambda_q} - 1. \tag{6.21}$$

When $q = p$ (i.e., $\lambda_q = \lambda_p$), the KL divergence is zero as expected.

6.6.4 Kumaraswamy-Uniform KL

The Kumaraswamy distribution provides a tractable encoder for latent variables on bounded domains. For $q = \text{Kum}(a, b)$ and the uniform prior $p = \mathcal{U}(0, 1)$, we obtain:

$$\text{KL}(q\|p) = \log(ab) - \left(1 - \frac{1}{a}\right)(\Psi(b+1) + \gamma) - \left(1 - \frac{1}{b}\right), \tag{6.22}$$

where Ψ is the digamma function and γ is the Euler–Mascheroni constant.

6.6.5 Wrapped Normal-Uniform KL

The Wrapped Normal distribution $q = \text{WN}(\mu, \sigma^2)$ on S^1 is constructed by wrapping a Gaussian onto the circle:

$$z = \arctan 2(\sin(\mu + \sigma\epsilon), \cos(\mu + \sigma\epsilon)), \tag{6.23}$$

where $\epsilon \sim \mathcal{N}(0, 1)$. This construction inherits exact reparametrization from the Gaussian.

The KL divergence between $q = \text{WN}(\mu, \sigma^2)$ and the circular uniform prior $p = \mathcal{U}(S^1)$ admits a closed-form approximation. Since the uniform prior has constant density $p(z) = 1/(2\pi)$, the KL divergence simplifies to

$$\text{KL}(q\|p) = \int q(z) \log q(z)\, dz - \int q(z) \log p(z)\, dz \tag{6.24}$$

$$= -H(q) + \log(2\pi), \tag{6.25}$$

where $H(q)$ denotes the differential entropy of the Wrapped Normal. For small σ, the wrapping has negligible effect and q is approximately Gaussian. Using the Gaussian entropy $H(\mathcal{N}(\mu, \sigma^2)) = \frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2}$, we obtain

$$\text{KL}(q\|p) \approx \log(2\pi) - \frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2} \tag{6.26}$$

$$= \frac{1}{2}\left(\log(2\pi) - \log(\sigma^2) - 1\right). \tag{6.27}$$

For large σ, the wrapped distribution approaches uniform on the circle, so $H(q) \to \log(2\pi)$ and $\mathrm{KL}(q\|p) \to 0$. We combine these regimes by clamping:

$$\mathrm{KL}(q\|p) \approx \max\left(0, \frac{1}{2}\left(\log(2\pi) - \log(\sigma^2) - 1\right)\right). \tag{6.28}$$

This approximation is nearly exact for $\sigma \lesssim 0.5$, where the wrapping has negligible effect. Its error slightly grows through the intermediate regime and peaks at roughly 0.10 near $\sigma = 1.5$, beyond which the approximation returns zero due to clamping while the true divergence remains positive and decays rapidly. When this residual matters, numerically integrating the Fourier series of the wrapped density recovers the divergence to machine precision at the cost of additional computation.

6.6.6 von Mises-Fisher-Uniform KL

For a hyperspherical factor S^{n-1} with $q = \mathrm{vMF}(\mu, \kappa)$ and the uniform prior $p = \mathcal{U}(S^{n-1})$, the KL divergence is available in closed form [139, App. B]:

$$\mathrm{KL}(q\|p) = \kappa \frac{I_{n/2}(\kappa)}{I_{n/2-1}(\kappa)} + \log C_n(\kappa) + \log \frac{2\pi^{n/2}}{\Gamma(n/2)}, \tag{6.29}$$

where I_ν is the modified Bessel function of the first kind of order ν, the normalizing constant of the vMF density is $C_n(\kappa) = \kappa^{n/2-1} / \left((2\pi)^{n/2} I_{n/2-1}(\kappa)\right)$, and $2\pi^{n/2}/\Gamma(n/2)$ is the surface area of S^{n-1}. The divergence depends on the concentration κ only, since the uniform prior is invariant to the mean direction μ.

6.6.7 Möbius Strip KL Divergence

The Möbius strip $M = (S^1 \times [0,1])/\mathbb{Z}_2$ is the quotient of the cylinder $C = S^1 \times [0,1]$ by the action $\tau(h, \theta) = (1 - h, \theta + \pi)$. This action identifies opposite edges with a twist, creating the Möbius topology. The encoder distribution q_C lives on the cylinder C, while the uniform prior p_M lives on M. We derive the KL divergence $\mathrm{KL}(q_M\|p_M)$, where q_M is the pushforward of q_C under the quotient map.

Let p_C denote the uniform distribution on the cylinder. Since M has half the area of C, we have $p_M([z]) = 2p_C(z)$ for each equivalence class $[z] \in M$. The encoder density at $[z] \in M$ is $q_M([z]) = q_C(z) + q_C(\tau(z))$, summing contributions from both identified points.

The KL divergence on M is given by

$$\begin{aligned}
\mathrm{KL}(q_M\|p_M) &= \int_M q_M([z]) \log \frac{q_M([z])}{p_M([z])}\, d[z] \\
&= \int_C q_C(z) \log \frac{q_C(z) + q_C(\tau(z))}{2p_C(z)}\, dz \\
&= \mathrm{KL}(q_C\|p_C) - \log 2 + \mathbb{E}_{z\sim q_C}[\log(1 + r(z))],
\end{aligned} \tag{6.30}$$

where $r(z) := q_C(\tau(z))/q_C(z)$ is the density ratio at identified points. The correction term $-\log 2 + \mathbb{E}[\log(1+r)]$ accounts for the quotient structure. When the encoder concentrates mass at one of the identified points only (occupying half the cylinder), we obtain $r \to 0$ and the correction approaches $-\log 2$, reflecting that M has half the volume of C. When the encoder places equal mass on both identified points ($r = 1$), the correction vanishes. In practice, we compute the density ratio $r(z)$ in log-space for numerical stability, using the log-probabilities from the Wrapped Normal and Kumaraswamy distributions. The expectation $\mathbb{E}[\log(1+r)]$ is approximated via Monte Carlo sampling from q_C.

6.7 Appendix: Constructing Feature Map ρ Using the Reynolds Operator

Section 6.3.3 introduced the Reynolds operator R for constructing G-invariant feature maps ρ. Here we give the step-by-step construction procedure and work through the Möbius strip derivation in full.

To construct an orbit-separating feature map ρ for a quotient $\mathcal{Z} = \widetilde{\mathcal{Z}}/G$, proceed as follows:

1. Choose a basis of coordinate functions for the covering space. For each factor of $\widetilde{\mathcal{Z}}$, select functions $f_1, \ldots, f_m$ that naturally parameterize it:
   - *Circular factor* S^1*:* use $\{\cos\theta, \sin\theta\}$.
   - *Interval factor* $[0,1]$*:* use $\{h - \frac{1}{2}\}$.
   - *Unbounded factor* $\mathbb{R}$ *or* $[0,\infty)$*:* use $\{z\}$ (or $\{z-c\}$ centered at a symmetry-compatible point c).

   The full basis for the covering space is the union of these per-factor functions. Good choices make the group action act simply (e.g., as sign flips) on as many basis elements as possible, which simplifies the subsequent steps.
2. Form monomials in $f_1, \ldots, f_m$ up to some degree d.
3. Apply R to each monomial.
4. Discard any result that vanishes identically, and remove linearly dependent features.
5. Verify that the surviving features separate G-orbits (i.e., $\rho(z_1) = \rho(z_2)$ implies $z_1 \sim z_2$). If not, increase d and repeat.

Noether's degree bound [165, Thm. 2.1.4] guarantees termination at $d \leq |G|$. All components of ρ are polynomials or trigonometric polynomials by construction, so they are smooth and their gradients are available in closed form for backpropagation.

6.7.1 Application to the Möbius Strip

The Möbius strip is $([0,1]\times S^1)/\mathbb{Z}_2$ with deck transformation $\tau(h,\theta)=(1-h,\theta+\pi)$. Following Step 1 of the procedure, the covering space is a cylinder $[0,1]\times S^1$, so we select $f_1=\cos\theta$, $f_2=\sin\theta$ for the circular factor and $f_3=h-\frac{1}{2}$ for the interval factor (centering at $\frac{1}{2}$ ensures that τ acts as a sign flip on f_3). Under τ, all three basis elements flip sign: $\cos\theta\mapsto-\cos\theta$, $\sin\theta\mapsto-\sin\theta$, and $h-\frac{1}{2}\mapsto-(h-\frac{1}{2})$.

Since each basis element is odd under τ, the Reynolds operator applied to any degree-1 monomial vanishes: $R[\cos\theta]=\frac{1}{2}(\cos\theta+(-\cos\theta))=0$, and likewise for $\sin\theta$ and $h-\frac{1}{2}$. At degree 2, products of two odd functions are even, so they survive. The non-redundant degree-2 invariants are:

$$\begin{aligned}
R[\cos^2\theta] &= \cos^2\theta=\tfrac{1}{2}(1+\cos 2\theta),\\
R[\sin^2\theta] &= \sin^2\theta=\tfrac{1}{2}(1-\cos 2\theta),\\
R[\cos\theta\cdot\sin\theta] &= \cos\theta\sin\theta=\tfrac{1}{2}\sin 2\theta,\\
R[(h-\tfrac{1}{2})^2] &= (h-\tfrac{1}{2})^2,\\
R[\cos\theta\cdot(h-\tfrac{1}{2})] &= \cos\theta\cdot(h-\tfrac{1}{2}),\\
R[\sin\theta\cdot(h-\tfrac{1}{2})] &= \sin\theta\cdot(h-\tfrac{1}{2}).
\end{aligned}$$

The first three are linearly dependent with $\{\cos 2\theta,\sin 2\theta,1\}$ (and the constant 1 carries no information), leaving the five independent features (6.13).

To verify orbit separation: from $\cos 2\theta$ and $\sin 2\theta$, the angle θ is determined up to θ or $\theta+\pi$. If $\theta_2=\theta_1$, the cross terms force $h_2=h_1$ (identical points). If $\theta_2=\theta_1+\pi$, the cross terms force $h_2=1-h_1$, which is exactly the $\mathbb{Z}_2$ identification. So ρ is injective on orbits.

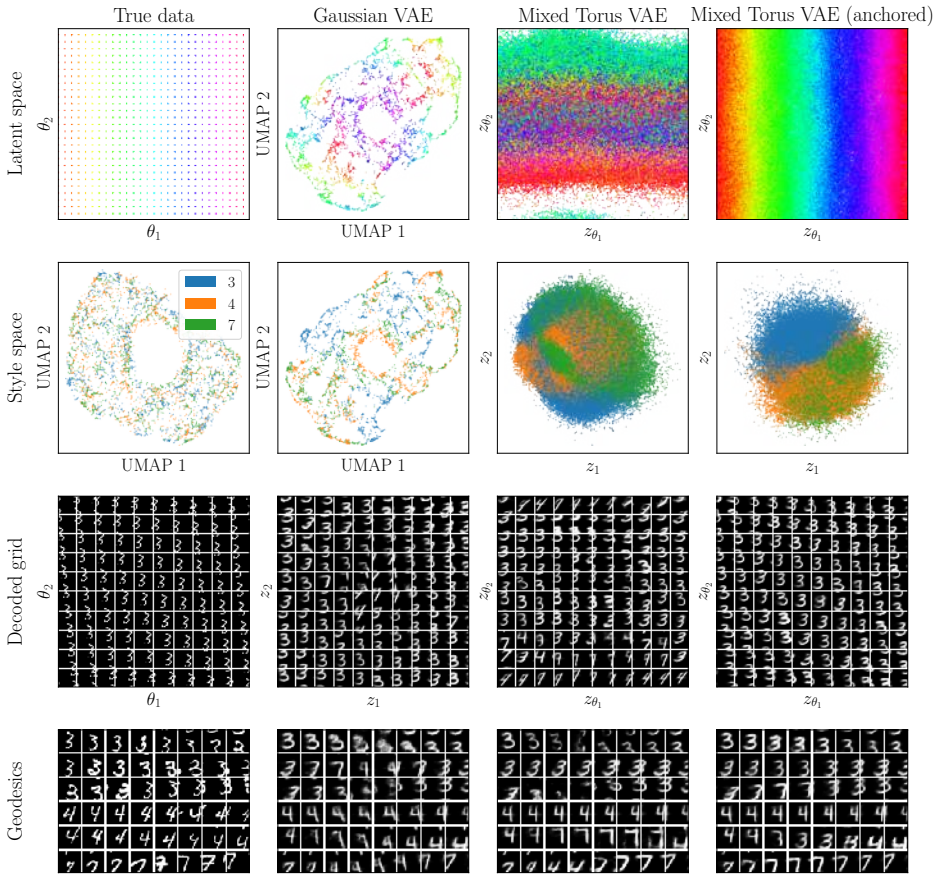


Figure 6.7. Shifted MNIST experiment ($\mathbb{R}^2 \times \mathbb{T}^2$ latent space, digit classes 3, 4, and 7). Columns: ground truth, Gaussian VAE ($\ell = 4$), Mixed-Torus VAE, and Mixed-Torus VAE (anchored). First row: latent coordinates colored by the true shift angle θ_1. The ground-truth column plots (θ_1, θ_2) directly. The Mixed-Torus VAEs plot the learned circular coordinates $(z_{\theta_1}, z_{\theta_2})$. The Gaussian column shows a UMAP projection of the full latent space. Second row: latent coordinates colored by digit class. The Mixed-Torus VAEs show the Euclidean style coordinates (z_1, z_2). The ground-truth and Gaussian columns show a UMAP projection. Third row: 10×10 decoded grid, shown for digit 3. The Mixed-Torus VAEs sweep both circular coordinates $(z_{\theta_1}, z_{\theta_2})$ with style coordinates fixed at the digit-3 centroid. The Gaussian VAE sweeps its first two Euclidean dimensions (z_1, z_2), with the remaining dimensions fixed at the digit-3 centroid. Fourth row: decoded geodesic paths, one per digit class, with style coordinates fixed at the class centroid.

7

Estimating Evolving Functions with Dynamic Gaussian Processes

Abstract - This chapter develops the Dynamic Gaussian Process (DGP), a framework for estimating continuous functions that evolve in discrete time and are observed through sparse, noisy measurements. Such functions are modeled by integro-difference equations (IDEs), which arise naturally from the time-discretization of linear partial differential equations (PDEs). We show that the framework generalizes both standard Gaussian process regression to time-varying functions and Kalman filtering to infinite-dimensional states. The DGP produces a Gaussian posterior over the evolving function, with mean and covariance functions available in closed form. When the kernels have a separable structure, this infinite-dimensional posterior is represented exactly by a finite-dimensional Kalman filter on basis function coefficients. For functions that do not lie in the span of the basis, the finite-dimensional filter is an approximation. We present an analysis of the induced L_2 estimation error, which decomposes exactly into a noise-limited error, a leakage-induced gap from the finite-dimensional approximation, and an out-of-subspace residual, all computable from the model. The latter two vanish as the number of basis functions M grows at a rate set by the kernel, so that the estimator recovers the exact infinite-dimensional posterior in the limit. At finite M, the filter's reported covariance omits the leakage-induced gap and is therefore overconfident, but the true error covariance

This chapter is based on van Hulst et al. [171, 172].

is computable from the model. Vector-valued states extend the framework to linear PDEs of higher temporal order, such as the wave equation. The estimator is demonstrated on the heat and wave equations. Code is available at `https://github.com/JvHulst/Dynamic_Gaussian_Processes`.

7.1 Introduction

Estimating evolving continuous functions from sparse, noisy measurements appears in many applications, including heat flow, battery thermal management, fluid dynamics, and population dynamics. Standard estimation methods, however, either address estimation in finite-dimensional spaces (e.g., Kalman filtering) or estimation of static functions (e.g., Gaussian process regression). This chapter investigates estimation problems for a class of evolving function models that relaxes both assumptions simultaneously.

The considered class of evolving function models is based on integro-difference equations (IDE), which model continuous functions that evolve with discrete-time dynamics. IDEs are spatio-temporal models with both a time dimension and a space dimension [59, 173, 174]. IDE models are currently employed in population evolution modeling [175], geographical processes [176, 177], and can be connected to the estimation of finite-dimensional time-discretized partial differential equations [59, Chapter 6]. Estimating such a field from sparse and noisy measurements, while propagating it through known dynamics, is the problem of data assimilation [178, 179] and, more broadly, of Bayesian inference over function-valued states [180].

Two natural starting points for this estimation problem are the Kalman filter (KF) [115], the quintessential estimator for finite-dimensional linear dynamical systems, and Gaussian process (GP) regression [78], a widely used non-parametric method for estimating static functions. The KF and the GP are closely related: both assume Gaussian noise and perform Bayesian conditioning, and spatio-temporal GP regression can be cast as infinite-dimensional Bayesian filtering solved through Kalman recursions [181]. This connection also yields efficient spatio-temporal Gaussian regression, in which a separable space-time covariance with a rational temporal spectrum is represented as a temporal Kalman filter [182, 183]. Reduced-rank expansions of the covariance function keep such GP models computationally tractable [184, 185], and the same basis function reduction underlies dimension-reduced filtering of spatio-temporal fields, as in fixed-rank filtering [186] and the Kriged Kalman Filter (KKF) [187]. All of these methods reduce the infinite-dimensional estimator to a finite computation, an approximation whose error is analyzed in Section 7.5.

At the center of this chapter is the Dynamic Gaussian Process (DGP), which unifies the KF and the GP, and was shown in our previous conference paper to be the exact Bayesian estimator for the IDE model class [171]. The key insight is that a GP remains a GP after both IDE evolution and conditioning on observations, so the DGP posterior admits closed-form mean and covariance updates. When the kernels have a separable structure (or are approximated as such), this infinite-dimensional posterior is represented by a finite-dimensional Kalman filter on basis function coefficients. The DGP is built from the evolution dynamics rather than a

prescribed covariance, and its finite-dimensional representation generalizes [176] and is structurally related to the KKF, as discussed in Section 7.4.

This chapter extends the DGP in two directions. First, vector-valued states broaden the application domain to linear PDEs of higher temporal order, so that the wave equation (which is second-order in time) now falls within the same model class. Such distributed-parameter systems have a long estimation history [188], and the DGP brings them within the Gaussian-process framework. Second, we analyze the approximation error of the finite-dimensional reduction. When the field leaves the span of the basis, the functional estimation error decomposes exactly into a noise-limited term, a leakage-induced gap, and an out-of-subspace residual, all computable from the model. The latter two vanish as the number of basis functions M grows, at a rate set by the smoothness of the kernel, so the reduced estimator converges to the exact Bayesian posterior. Because this error is computable, the filter's overconfident uncertainty can be corrected to the true value.

The rest of this chapter is organized as follows. Section 7.2 formulates the IDE model class, its connection to PDEs, and provides relevant background on the KF and the GP. Section 7.3 presents the exact DGP solution. Section 7.4 shows how separable kernel structure reduces the DGP to a finite-dimensional Kalman filter and discusses connections to existing frameworks. Section 7.5 analyzes the stability, error, and convergence of the finite-dimensional approximation. Section 7.6 presents numerical examples for the heat and wave equations, and Section 7.7 gives conclusions.

Notation. Let $\mathbb{R} = (-\infty, \infty)$, and $\mathbb{R}_{\geq 0} = [0, \infty)$. Let $\mathbb{N} = \{0, 1, \ldots\}$ denote the natural numbers. The identity matrix of size n is denoted by I_n. The Kronecker product is denoted $\otimes$. Let $\mathbb{S}_+^n := \{A \in \mathbb{R}^{n \times n} | A \succeq 0\}$ denote the set of symmetric positive semidefinite matrices of size n. A normal distribution with mean vector $\mu \in \mathbb{R}^n$ and covariance matrix $\Sigma \in \mathbb{S}_+^n$ is denoted $\mathcal{N}(\mu, \Sigma)$. A Gaussian process with mean function $\bar{f}(x)$ and covariance function $k(x, x')$ is denoted $\mathcal{GP}(\bar{f}(x), k(x, x'))$.

Given a function $f : \mathcal{X} \to \mathbb{R}$ and a vector $X = [x_1, x_2, \ldots, x_n]^\top \in \mathcal{X}^n$, we use the following compact notation:

$$\mathbf{f}(X) := [f(x_1), f(x_2), \ldots, f(x_n)]^\top \in \mathbb{R}^n.$$

Similarly, for a kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and vectors $X \in \mathcal{X}^n$, $X' \in \mathcal{X}^m$, the kernel matrix $\mathbf{k}(X, X') \in \mathbb{R}^{n \times m}$ is defined entrywise by

$$\mathbf{k}(X, X')_{ij} := k(x_i, x'_j),$$

for $1 \leq i \leq n$ and $1 \leq j \leq m$. Throughout, boldface denotes evaluation of a function or kernel at a vector of spatial points. When $f : \mathcal{X} \to \mathbb{R}^D$ is vector-valued with components $f^{(1)}, \ldots, f^{(D)}$, the boldface convention extends by stacking output dimensions,

$$\mathbf{f}(X) := [\mathbf{f}^{(1)\top}(X), \ldots, \mathbf{f}^{(D)\top}(X)]^\top \in \mathbb{R}^{Dn},$$

and similarly for matrix-valued kernels $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$, where $\mathbf{K}(X, X') \in \mathbb{R}^{Dn\times Dm}$ follows the same dimension-first ordering.

7.2 Problem Formulation and Preliminaries

This section introduces the integro-difference equation model class and its connection to partial differential equations, followed by the observation model and the relevant background on Gaussian process regression and Kalman filtering.

7.2.1 Integro-Difference Equations

Consider a function $f_t : \mathcal{X} \to \mathbb{R}^D$ defined on a spatial domain $\mathcal{X}$ that evolves in discrete time according to an integro-difference equation (IDE),

$$f_{t+1}(x) = \int_{\mathcal{X}} K_f(x, s)\, f_t(s)\, d\nu(s) + v_t(x), \tag{7.1}$$

with $x \in \mathcal{X}$, $t \in \mathbb{N}$, state function $f_t : \mathcal{X} \to \mathbb{R}^D$, matrix-valued evolution kernel $K_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$, and stochastic disturbance $v_t(x)$. Here, ν is a σ-finite reference measure on $\mathcal{X}$, typically the Lebesgue measure for spatially continuous systems or the counting measure when $\mathcal{X}$ is a finite set. IDEs of this form model continuous functions that evolve with discrete-time dynamics. They are spatio-temporal models with many practical applications [59, 173, 175–177]. For $D = 1$, the kernel K_f reduces to a scalar function $k_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and (7.1) is the standard scalar IDE.

Let $\mathcal{K}$ denote the integral operator associated with K_f, defined by

$$(\mathcal{K}f)(x) := \int_{\mathcal{X}} K_f(x, s)\, f(s)\, d\nu(s), \tag{7.2}$$

so that (7.1) can be written compactly as $f_{t+1} = \mathcal{K}f_t + v_t$. We assume that $f_0(x) \sim \mathcal{GP}(\bar{f}_0(x), Q_f(x, x'))$ in which $\bar{f}_0 : \mathcal{X} \to \mathbb{R}^D$ and $Q_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$ are the mean function and positive semidefinite matrix-valued covariance function of the initial condition, respectively. The disturbances are $v_t(x) \sim \mathcal{GP}(0, Q_v(x, x'))$, $t \in \mathbb{N}$, with $Q_v : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$ positive semidefinite and independent of f_t.

A natural source of IDE models is the time-discretization of partial differential equations [59, Sections 7.2.4 and 7.2.5]. Consider a linear PDE of temporal order $r \geq 1$ on a domain $\mathcal{X} \subseteq \mathbb{R}^d$, subject to boundary conditions on $\partial\mathcal{X}$ such that the initial-boundary value problem is well-posed:

$$\frac{\partial^r \phi}{\partial \tau^r}(x, \tau) = \mathcal{L}\,\phi(x, \tau), \tag{7.3}$$

where $\phi : \mathcal{X} \times \mathbb{R}_{\geq 0} \to \mathbb{R}$, $\tau \in \mathbb{R}_{\geq 0}$ is the continuous-time variable, and $\mathcal{L}$ is a linear spatial differential operator. The solution of (7.3) depends on r initial conditions $\phi(x,0), \partial_\tau \phi(x,0), \ldots, \partial_\tau^{r-1}\phi(x,0)$. Collecting these into a vector-valued state $f(x,\tau) \in \mathbb{R}^D$ with $D \leq r$ through the standard state-space reduction, the time evolution from 0 to τ is described by a bounded linear solution operator $T(\tau) : L_2(\mathcal{X};\mathbb{R}^D) \to L_2(\mathcal{X};\mathbb{R}^D)$ [189, Chapters 4, 7]:

$$f(\cdot,\tau) = T(\tau)\, f(\cdot,0). \tag{7.4}$$

If the solution operator $T(\tau)$ admits an integral kernel $\mathcal{S}(x,s,\tau) \in \mathbb{R}^{D\times D}$, then (7.4) can be written as

$$f(x,\tau) = \int_{\mathcal{X}} \mathcal{S}(x,s,\tau)\, f(s,0)\, ds. \tag{7.5}$$

The kernel $\mathcal{S}$ is the Green's function of the PDE, with the matrix-valued structure arising from the state augmentation when $r > 1$. Existence and regularity of Green's functions for linear PDEs, including the parabolic and hyperbolic classes considered in this chapter, is classical [190, Chapters 2, 7]. Section 7.6 derives the kernels in closed form for the heat equation ($D = 1$) and the wave equation ($D = 2$).

To transition from this continuous dynamic to a discrete-time filtering formulation, we fix a sampling interval $\Delta > 0$. The semigroup property $T(\tau+\Delta) = T(\Delta)\,T(\tau)$ implies $f(x,\tau+\Delta) = \int_{\mathcal{X}} \mathcal{S}(x,s,\Delta)\, f(s,\tau)\, ds$ for any $\tau \geq 0$. Writing $f_t(x) := f(x, t\Delta)$ for $t \in \mathbb{N}$ then gives

$$f_{t+1}(x) = \int_{\mathcal{X}} \mathcal{S}(x,s,\Delta)\, f_t(s)\, ds, \tag{7.6}$$

which is an instance of (7.1) with $K_f(x,s) = \mathcal{S}(x,s,\Delta)$.

Remark 7.1. *The IDE* (7.6) *is a consistent time discretization of the PDE* (7.3)*: setting* $v_t = 0$ *and letting* $\Delta \to 0$*,*

$$\frac{(\mathcal{K}f)(x) - f(x)}{\Delta} \to (\mathcal{L}f)(x) \tag{7.7}$$

for all f *in the domain of* $\mathcal{L}$*, recovering the original PDE dynamics. This follows from standard semigroup theory: the solution operators* $\{T(\tau)\}_{\tau\geq 0}$ *form a strongly continuous (*C_0*-)semigroup whose infinitesimal generator is the spatial differential operator [189, Chapter 1]. For higher-order PDEs (*$r > 1$*), the state augmentation produces a first-order system on* $L_2(\mathcal{X};\mathbb{R}^D)$*, to which the same semigroup argument applies with the generator acting on the product space.*

The PDE connection above provides one natural source of IDE models, but the formulation in (7.1) is more general. The evolution kernel K_f can also be specified

directly. While the results in this chapter hold for general kernel functions K_f, Q_f, Q_v, they will lead to particularly efficient implementations when the kernels take a special separable form. A matrix-valued kernel $K(x, x') \in \mathbb{R}^{D\times D}$ is called separable if each of its entries satisfies

$$K_{ij}(x, x') = U^\top(x)\Lambda_{ij}U(x'), \quad i, j \in \{1, \dots, D\}, \tag{7.8}$$

for a shared vector of basis functions

$$U(x) := [u_1(x), \dots, u_M(x)]^\top \in \mathbb{R}^M$$

and coefficient matrices $\Lambda_{ij} \in \mathbb{R}^{M\times M}$. Equivalently, defining $\Lambda := [\Lambda_{ij}]_{i,j=1}^D \in \mathbb{R}^{DM\times DM}$ and writing $\check{U}(x) := I_D \otimes U(x) \in \mathbb{R}^{DM\times D}$ gives the compact form $K(x, x') = \check{U}^\top(x)\,\Lambda\,\check{U}(x')$. For $D = 1$, $\check{U}(x) = U(x)$ and this reduces to $k(x, x') = U^\top(x)\Lambda\,U(x')$. The benefits of this structure are further examined in Section 7.4.

7.2.2 Observations and Estimation Problem

At each time step t, we obtain p scalar observations $Y_t \in \mathbb{R}^p$ of the function f_t at spatial locations $X_t \in \mathcal{X}^p$, according to

$$Y_t = \Phi_t\,\mathbf{f}_t(X_t) + \mathbf{w}_t(X_t), \tag{7.9}$$

where $\mathbf{f}_t(X_t) \in \mathbb{R}^{Dp}$ stacks the state evaluations by output dimension, and $\Phi_t \in \mathbb{R}^{p\times Dp}$ is a known observation matrix that selects which state components are measured. The measurement noise is $w_t(x) \sim \mathcal{GP}(0, Q_w(x, x'))$, $t \in \mathbb{N}$, with positive semidefinite $Q_w : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. For $D = 1$, the observation matrix reduces to $\Phi_t = I_p$ and (7.9) recovers the standard observation model $Y_t = \mathbf{f}_t(X_t) + \mathbf{w}_t(X_t)$. For $D > 1$, the matrix Φ_t enables partial state observation; for instance, if only the first of two state components is measured, then $\Phi_t = [I_p \;\; 0_p] \in \mathbb{R}^{p\times 2p}$.

The problem considered in this chapter is to characterize the posterior distribution of f_t given the data $\{X_{0:t}, Y_{0:t}\}$ recursively for each $t \in \mathbb{N}$, where $X_{0:t} := [X_0, \dots, X_t]^\top$ and $Y_{0:t} := [Y_0, \dots, Y_t]^\top$ as in (7.9). Besides enabling the estimation of time-discretized PDEs of the form (7.3), this problem can be motivated from two additional perspectives:

1. as an extension of GP regression to the case where the function evolves in time;

2. as an extension of Kalman filtering to infinite-dimensional systems described by integro-difference equations.

These connections are addressed in the next subsections, which also provide preliminaries for the remainder of the chapter.

7.2.3 Gaussian Process Regression

A Gaussian process (GP) is a distribution over the space of functions, fully specified by a prior mean function $\bar{f}(x) = \mathbb{E}[f(x)]$ and a covariance (kernel) function $k(x, x') = \text{cov}(f(x), f(x'))$ [78]. Given noisy observations $y_t = f(x_t) + w_t$ with $w_t \sim \mathcal{N}(0, \sigma^2)$ at inputs $X = \{x_0, \ldots, x_N\}$, the posterior distribution is again Gaussian:

$$\begin{aligned} \hat{f}(x) &= \bar{f}(x) + L(x, X)(Y - \bar{\mathbf{f}}(X)), \\ \hat{c}(x, x') &= k(x, x') - L(x, X)\mathbf{k}(X, x'), \end{aligned} \tag{7.10}$$

where $L(x, X) := \mathbf{k}(x, X)\left[\mathbf{k}(X, X) + \sigma^2 I_N\right]^{-1}$.

The standard GP is a special case of (7.1) with $D = 1$, $\mathcal{K} = \mathcal{I}$, and $Q_v = 0$, for which $f_{t+1} = f_t$. The estimation problem of Section 7.2.2 thus extends GP regression to functions that evolve in time.

7.2.4 Kalman Filtering

The Kalman filter is a well-known algorithm to estimate the state of a dynamical system described by

$$x_{t+1} = Ax_t + v_t, \tag{7.11}$$

$$y_t = C_t x_t + w_t, \tag{7.12}$$

where $x_t \in \mathbb{R}^n$ the state to be estimated, $y_t \in \mathbb{R}^p$ the system output, $v_t \sim \mathcal{N}(0, V)$, $t \in \mathbb{N}$ the disturbance on the state with covariance matrix $V \in \mathbb{S}_+^n$, and $w_t \sim \mathcal{N}(0, W_t)$, $t \in \mathbb{N}$, the measurement noise with covariance matrix $W_t \in \mathbb{S}_+^p$, with $x_0 \sim \mathcal{N}(\bar{x}_0, \bar{S})$. The output matrix C_t and the noise covariance W_t are chosen time-varying because in the IDE setting of Section 7.2.1, the observation locations X_t may differ at each time step.

Important for what follows is to detail the two steps that together constitute the Kalman filter, called the update and the prediction steps. The equations for the update step are given by

$$\begin{aligned} \hat{x}_{t|t} &= \hat{x}_{t|t-1} + L_t(y_t - C_t\hat{x}_{t|t-1}), \\ S_{t|t} &= S_{t|t-1} - L_t C_t S_{t|t-1}, \end{aligned} \tag{7.13}$$

with $L_t = S_{t|t-1}C_t^\top(C_t S_{t|t-1}C_t^\top + W_t)^{-1}$ the Kalman gain, $\hat{x}_{0|-1} = \bar{x}_0$ and $S_{0|-1} = \bar{S}$. Here, $\hat{x}_{t|l}$ denotes the estimate of x_t at time step l, and $S_{t|l}$ denotes the estimate of the state covariance $\text{cov}(x_t, x_t)$ at time step l. The equations for the prediction step in the KF are given by

$$\begin{aligned} \hat{x}_{t+1|t} &= A\hat{x}_{t|t}, \\ S_{t+1|t} &= AS_{t|t}A^\top + V. \end{aligned} \tag{7.14}$$

The mean and covariance estimates retain their linear and quadratic forms under both the update and the prediction steps. These steps use only the previous estimates and the current data, so the Kalman filter can be implemented recursively. Under linear dynamics and zero-mean Gaussian noise, it is the optimal estimator for this problem [115].

The standard Kalman filter is a special case of (7.1) and (7.9): for a finite set $\mathcal{X} = \{x_1, \dots, x_m\}$ with the counting measure, the integral in (7.1) reduces to a finite sum and the system becomes a linear state-space model with transition matrix $A \in \mathbb{R}^{Dm \times Dm}$ determined by the kernel evaluations $K_f(x_i, x_j)$.

The next section details the extension of the Kalman filter to infinite-dimensional systems modeled by IDEs, which presents the solution to the problem posed in Section 7.2.1.

7.3 Dynamic Gaussian Processes

Here, we consider the estimation problem of the system (7.1), which we refer to as the dynamic Gaussian process (DGP). The key results are two theorems showing that if the initial condition of the IDE in (7.1) is a GP, it remains a GP after evolving and after incorporating observations.

Theorem 7.2. *Suppose that* $f_0(x) \sim \mathcal{GP}(\bar{f}_0(x), Q_f(x, x'))$ *with* $\bar{f}_0 : \mathcal{X} \to \mathbb{R}^D$ *and* $Q_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$*, and that* $v_0(x) \sim \mathcal{GP}(0, Q_v(x, x'))$ *with* $Q_v : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$*, independent of* f_0*. Consider*

$$f_1(x) = \int_{\mathcal{X}} K_f(x, s) f_0(s)\, d\nu(s) + v_0(x), \tag{7.15}$$

with $K_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$*. Then* $f_1(x)$ *is a GP with mean* $\bar{f}_1 : \mathcal{X} \to \mathbb{R}^D$ *and covariance* $Q_1 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D \times D}$ *given by*

$$\bar{f}_1(x) = \int_{\mathcal{X}} K_f(x, s) \bar{f}_0(s)\, d\nu(s), \tag{7.16}$$

$$Q_1(x, x') = \int_{\mathcal{X}} \int_{\mathcal{X}} K_f(x, s)\, Q_f(s, s')\, K_f^\top(x', s') d\nu(s)\, d\nu(s') + Q_v(x, x'). \tag{7.17}$$

Proof. The mean of $f_1(x)$ follows from linearity of expectation and the integral:

$$\begin{aligned} \mathbb{E}[f_1(x)] &= \int_{\mathcal{X}} K_f(x, s)\, \mathbb{E}[f_0(s)]\, d\nu(s) + \mathbb{E}[v_0(x)] \\ &= \int_{\mathcal{X}} K_f(x, s)\, \bar{f}_0(s)\, d\nu(s). \end{aligned}$$

For the covariance, expand $\mathbb{E}\big[(f_1(x)-\bar{f}_1(x))(f_1(x')-\bar{f}_1(x'))^\top\big]$. Independence of f_0 and v_0 removes the cross terms, $\mathbb{E}[v_0(x)\, v_0^\top(x')] = Q_v(x,x')$ gives the second term of (7.17), and $\mathbb{E}\big[(f_0(s)-\bar{f}_0(s))(f_0(s')-\bar{f}_0(s'))^\top\big] = Q_f(s,s')$ gives the first. Finally, f_1 is a GP because any finite collection of its evaluations is a linear transformation of the jointly Gaussian f_0 and v_0. □

With the prediction step established, the remaining ingredient is conditioning the GP on observations.

Theorem 7.3. *Suppose that $f_0(x) \sim \mathcal{GP}(\bar{f}_0(x), Q_f(x,x'))$ with $\bar{f}_0 : \mathcal{X} \to \mathbb{R}^D$ and $Q_f : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$, and that $w_0(x) \sim \mathcal{GP}(0, Q_w(x,x'))$ is independent of f_0. Consider the observations*

$$Y_0 = \Phi_0\, \mathbf{f}_0(X_0) + \mathbf{w}_0(X_0), \tag{7.18}$$

with $\Phi_0 \in \mathbb{R}^{p\times Dp}$ the observation matrix. Conditioning the GP f_0 on Y_0 yields another GP with mean

$$\bar{f}_0(x) + \mathbf{Q_f}(x, X_0)\Phi_0^\top \Sigma_0^{-1}(Y_0 - \Phi_0\, \bar{\mathbf{f}}_0(X_0)), \tag{7.19}$$

and covariance

$$Q_f(x,x') - \mathbf{Q_f}(x, X_0)\Phi_0^\top \Sigma_0^{-1}\, \Phi_0\, \mathbf{Q_f}(X_0, x'), \tag{7.20}$$

in which $\Sigma_0 := \Phi_0\, \mathbf{Q_f}(X_0, X_0)\, \Phi_0^\top + \mathbf{Q_w}(X_0, X_0)$ is the covariance of Y_0, and $\mathbf{Q_f}(x, X_0) \in \mathbb{R}^{D\times Dp}$ denotes the cross-covariance between $f_0(x)$ and the stacked evaluations $\mathbf{f}_0(X_0)$.

Proof. Since f_0 and w_0 are independent GPs, Y_0 and $f_0(x)$ are jointly Gaussian for every x, with $\mathrm{cov}(Y_0) = \Sigma_0$ and cross-covariance $\mathbf{Q_f}(x, X_0)\, \Phi_0^\top$. The standard Gaussian conditioning formula then gives the stated mean and covariance, and since x is arbitrary the conditional law is a GP. □

Alternating the prediction step of Theorem 7.2 and the update step of Theorem 7.3 yields the DGP estimator.

7.3.1 DGP Estimator

We aim to estimate the evolving function $f_t(x) \in \mathbb{R}^D$ using the model (7.1) and observation model (7.9). The estimate is a GP characterized by a mean function $\hat{f}_{t|l} : \mathcal{X} \to \mathbb{R}^D$ and a covariance function $\hat{c}_{t|l} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{D\times D}$. The double subscripts indicate that we estimate f_t using information up to time step l. Prior knowledge is embedded through the initial conditions $\hat{f}_{0|-1} = \bar{f}_0$ and $\hat{c}_{0|-1} = Q_f$.

The update step, based on noisy measurements Y_t at spatial locations X_t, is given by:

$$\begin{aligned}\hat{f}_{t|t}(x) &= \hat{f}_{t|t-1}(x) + L_t(x, X_t)(Y_t - \Phi_t \hat{\mathbf{f}}_{t|t-1}(X_t)),\\ \hat{c}_{t|t}(x, x') &= \hat{c}_{t|t-1}(x, x') - L_t(x, X_t)\, \Phi_t\, \hat{\mathbf{c}}_{t|t-1}(X_t, x'),\end{aligned} \tag{7.21}$$

in which

$$L_t(x, X_t) := \hat{\mathbf{c}}_{t|t-1}(x, X_t)\, \Phi_t^\top \left[\Phi_t\, \hat{\mathbf{c}}_{t|t-1}(X_t, X_t)\, \Phi_t^\top + \mathbf{Q}_{\mathbf{w}}(X_t, X_t)\right]^{-1}.$$

The prediction step is given by

$$\begin{aligned}\hat{f}_{t+1|t}(x) &= \int_{\mathcal{X}} K_f(x, s)\, \hat{f}_{t|t}(s)\, d\nu(s),\\ \hat{c}_{t+1|t}(x, x') &= \int_{\mathcal{X}}\int_{\mathcal{X}} K_f(x, s)\, \hat{c}_{t|t}(s, s')\, {K_f}^\top(x', s') d\nu(s)\, d\nu(s') + Q_v(x, x').\end{aligned} \tag{7.22}$$

While the DGP updates (7.21)–(7.22) are exact, their direct implementation requires integrals over the full domain at every prediction step, and these integrals rarely admit analytical solutions. Moreover, the posterior mean and covariance expressions grow more complicated at each time step. The next section shows that this cost can be managed when the kernels have a separable structure.

7.4 Separable Kernels

In this section, we show that when K_f, Q_f, and Q_v are separable kernels we can perform the required computations of the previous section efficiently. We start with the DGP with truly separable kernels in Section 7.4.1, and show in Section 7.4.2 that separable kernels can be used to approximate the problem from Section 7.2.1.

7.4.1 DGP with Separable Kernels

To illustrate the reduced computational cost that follows from the separability of the kernels K_f, Q_f, and Q_v, consider the following lemmas.

Lemma 7.4. *Suppose that*

$$\begin{aligned}K_f(x, x') &= \check{U}^\top(x)\, \Lambda\, \check{U}(x'),\\ Q_f(x, x') &= \check{U}^\top(x)\, \Lambda_f\, \check{U}(x'),\\ Q_v(x, x') &= \check{U}^\top(x)\, \Lambda_v\, \check{U}(x')\end{aligned} \tag{7.23}$$

with $\Lambda \in \mathbb{R}^{DM \times DM}$, $\Lambda_f, \Lambda_v \in \mathbb{S}_+^{DM}$ *and*

$$U(x) := [u_1(x), u_2(x), \ldots, u_M(x)]^\top$$

a vector of basis functions in which $u_i : \mathcal{X} \to \mathbb{R}$, *for* $i \in \{1, 2, \ldots, M\}$. *Define the transition matrix, observation matrix, and measurement noise covariance*

$$\begin{aligned} A_M &:= \Lambda \left(I_D \otimes \Lambda_U\right) \in \mathbb{R}^{DM \times DM}, \\ C_t &:= \Phi_t \, \check{\mathbf{U}}(X_t) \in \mathbb{R}^{p \times DM}, \\ W_t &:= \mathbf{Q}_{\mathbf{w}}(X_t, X_t) \in \mathbb{R}^{p \times p}, \end{aligned} \tag{7.24}$$

where $\check{\mathbf{U}}(X_t) := I_D \otimes \mathbf{U}(X_t) \in \mathbb{R}^{Dp \times DM}$ *denotes the block-diagonal basis evaluation at the observation locations* X_t *and* $\Lambda_U := \int_{\mathcal{X}} U(x) U^\top(x) \, d\nu(x) \in \mathbb{R}^{M \times M}$ *is the Gram matrix of the basis. Then the update step in* (7.21) *reduces to*

$$\hat{c}_{t|t}(x, x') = \check{U}^\top(x) \, \Psi_{t|t} \, \check{U}(x') \tag{7.25}$$

with

$$\Psi_{t|t} := \Psi_{t|t-1} - \Gamma_t \, C_t \, \Psi_{t|t-1}$$

in which $\Psi_{0|-1} = \Lambda_f$, *and*

$$\Gamma_t := \Psi_{t|t-1} \, C_t^\top \left[C_t \, \Psi_{t|t-1} \, C_t^\top + W_t\right]^{-1}.$$

Additionally, the prediction step in (7.22) *is reduced to*

$$\hat{c}_{t+1|t}(x, x') = \check{U}^\top(x) \, \Psi_{t+1|t} \, \check{U}(x') \tag{7.26}$$

with

$$\Psi_{t+1|t} := A_M \, \Psi_{t|t} \, A_M^\top + \Lambda_v.$$

Proof. Initialize $\hat{c}_{0|-1}(x, x') = \check{U}^\top(x) \, \Psi_{0|-1} \, \check{U}(x')$ with $\Psi_{0|-1} = \Lambda_f$, then substitute (7.23) into (7.21) and (7.22). The key identity is $\check{U}(s) \, \check{U}^\top(s) = I_D \otimes U(s) U^\top(s)$, which after integration yields $I_D \otimes \Lambda_U$. □

Remark 7.5. *Note that* Q_w *does not need to be approximated, as the measurement noise function* $w_t(x)$ *is always evaluated for a finite set of spatial locations* X_t *when taking observations according to* (7.9).

Under assumption (7.23), the covariance estimate is closed under both the update and the prediction steps. However, the mean estimate computation still scales poorly as the time index t increases. Efficient implementation of the mean estimate is enabled by the following lemma.

Lemma 7.6. *If* (7.23) *holds and additionally*

$$\bar{f}_0(x) = \check{U}^\top(x)\,\bar{z}, \tag{7.27}$$

with $\bar{z} \in \mathbb{R}^{DM}$ *and* $U(x)$ *the same vector of functions as in* (7.23)*, then the update of the mean estimate reduces to*

$$\hat{f}_{t|t}(x) = \check{U}^\top(x)\,z_{t|t}, \tag{7.28}$$

with

$$z_{t|t} := z_{t|t-1} + \Gamma_t(Y_t - C_t\,z_{t|t-1}).$$

Additionally, the prediction step in (7.22) *is reduced to*

$$\hat{f}_{t+1|t}(x) = \check{U}^\top(x)\,z_{t+1|t}, \tag{7.29}$$

with

$$z_{t+1|t} = A_M\,z_{t|t}.$$

Proof. Substitute (7.27) into (7.21) and (7.22). □

Note the similarity of the expressions in Lemma 7.6 to the standard KF. With Lemma 7.4 and Lemma 7.6, the DGP can be interpreted as a Kalman filter where the state vector $z_t \in \mathbb{R}^{DM}$ represents the coefficients of the function estimate projected onto the basis U. The matrices A_M, C_t, Λ_v, and W_t correspond directly to A, C_t, V, and W_t from Section 7.2.4. Each step costs $O((DM)^3)$, set by the basis size M rather than the spatial resolution or the length of the data record.

If both lemmas apply, the problem presented in Section 7.2 can be exactly estimated using the equations presented in this section. For the more general case, where (7.1) does not satisfy the lemmas, the problem can be approximated using the basis function methods detailed next.

7.4.2 Approximate DGP

In general, the function f_t does not lie exactly in $\operatorname{span}\{u_1, \ldots, u_M\}$. To handle this case, we define the coefficient representation of an arbitrary $f_t \in L_2(\mathcal{X}; \mathbb{R}^D)$ as the vector $z_t \in \mathbb{R}^{DM}$ satisfying $\check{U}^\top(x)z_t = (\pi_M f_t)(x)$, where π_M denotes the orthogonal projection onto $\operatorname{span}\{u_1, \ldots, u_M\}$. This generalizes the coefficient vector from Lemma 7.6, where f_t was assumed to lie exactly in the span of U.

The kernel functions K_f, Q_f, Q_v, and the prior mean $\bar{f}_0$ can be approximated in the separable form (7.23)–(7.27) by L_2-projection onto the basis U. This

approximation relates to reduced-order Kalman filtering [191], but with an infinite-dimensional full model. The resulting coefficient vector for the prior mean is

$$\bar{z}^* = (I_D \otimes \Lambda_U^{-1}) \int_{\mathcal{X}} \check{U}(x)\, \bar{f}_0(x)\, d\nu(x), \tag{7.30}$$

and the $DM \times DM$ coefficient matrix for a matrix-valued kernel $K \in \{K_f, Q_f, Q_v\}$ is

$$\Lambda_K^* = (I_D \otimes \Lambda_U^{-1})\, J_K\, (I_D \otimes \Lambda_U^{-1}), \tag{7.31}$$

where

$$J_K := \int_{\mathcal{X}}\int_{\mathcal{X}} \check{U}(x)\, K(x,x')\, \check{U}^\top(x')\, d\nu(x)\, d\nu(x').$$

For a separable kernel, Λ_K^* recovers its exact coefficient matrix, so the projections $\Lambda_{K_f}^*, \Lambda_{Q_f}^*, \Lambda_{Q_v}^*$ generalize the exact $\Lambda, \Lambda_f, \Lambda_v$ of Lemma 7.4. In practice, the required integrals can be computed approximately using Riemann sums, which only requires pointwise evaluations of K_f, Q_f, Q_v, $\bar{f}_0$.

For the minimizer to be unique, the basis functions must be linearly independent. If additionally the basis is orthonormal, i.e., $\langle u_i, u_j \rangle := \int_{\mathcal{X}} u_i(x) u_j(x)\, d\nu(x) = \delta_{ij}$, then $\Lambda_U = I_M$ and the projection decouples into independent scalar minimizations.

Common choices for U are the Fourier basis, piecewise-constant (bin) functions, radial basis functions, and hat functions. The approximation error introduced by the projection can be made arbitrarily small by increasing M, under mild conditions on the basis functions and the kernels. This is further analyzed in Section 7.5.

7.4.3 Connection to Existing Frameworks

The approximate DGP presented in this section is closely related to the work of [176], who also project the IDE estimation problem onto a set of basis functions. In that work, the basis U is required to be orthonormal, the initial condition f_0 is assumed to lie within the span of U, and the evolution kernel K_f is assumed to have the separable structure $K_f(x,x') = \tilde{U}^\top(x) U(x')$ for some unknown function vector $\tilde{U}(x)$. The observations Y_t are assumed to occur at the same spatial locations at every time step, with the number of measurement locations satisfying $p \geq M$. Under these assumptions, the coefficient dynamics reduce to a finite-dimensional state-space model that can be estimated using a standard Kalman filter. In contrast, the approximate DGP imposes none of these restrictions: the basis U need not be orthonormal, the spatial locations X_t may vary in time, and p may be smaller than M. The key reason is that the approximate DGP is derived from the exact DGP by projecting the posterior onto the basis, rather than by directly assuming a finite-dimensional model.

The approximate DGP also shares structural similarities with the Kriged Kalman Filter (KKF) [187]. In the KKF, a spatio-temporal field is decomposed into basis function coefficients that evolve with linear dynamics, and a spatially correlated residual that is estimated via kriging. Both the approximate DGP and the KKF propagate a finite-dimensional state vector $z_t \in \mathbb{R}^{DM}$ with a transition matrix and apply Kalman filter updates. However, the KKF transition matrix is specified directly, without connection to an evolution kernel K_f or to an underlying exact infinite-dimensional estimation problem. In the DGP, the transition matrix $A_M = \Lambda\,(I_D \otimes \Lambda_U)$ is derived from the evolution kernel through the L_2-projection, which provides a principled link between the finite-dimensional approximation and the exact problem.

A related line of work reaches similar Kalman recursions from the opposite starting point. Infinite-dimensional Bayesian filtering [181, 184] and efficient spatio-temporal Gaussian regression [182, 183] begin from a separable space-time covariance with a rational temporal spectrum, then recover a temporal state-space model by factorizing that covariance. In these frameworks the covariance is the modeling primitive and the dynamics follow from it. The DGP is specified in the reverse direction: the evolution operator $\mathcal{K}$ is the primitive, taken from the IDE or from the Green's function of a PDE, and the covariance of f_t follows from $\mathcal{K}$ and the disturbance covariance Q_v. A model given through $\mathcal{K}$ therefore need not have a stationary or rational-spectrum temporal covariance, as the heat and wave kernels of Section 7.6 show. The reduced DGP shares the separable-kernel structure (7.23) of these frameworks, so the difference lies in this dynamics-first starting point.

A further distinction is that the IDE model in (7.1) supports vector-valued states ($D > 1$), which enables the treatment of higher-order PDEs through state augmentation as described in Section 7.2.1. The DGP estimator inherits this generality, while the frameworks of [176, 181, 182, 186, 187] are formulated for scalar fields. In all of these frameworks, including the approximate DGP, the coefficient recursion is a standard reduced-rank Kalman filter. What distinguishes the present analysis is that it quantifies the error this reduction induces relative to the exact infinite-dimensional posterior, which is the subject of Section 7.5.

7.5 Stability and Approximation Error

The approximate DGP from Section 7.4.2 replaces the infinite-dimensional estimation problem with a Kalman filter on the coefficient vector $z_t \in \mathbb{R}^{DM}$, with transition matrix A_M, observation matrix C_t, and noise covariance W_t as defined in (7.24). For the steady-state analysis in this section, we assume stationary measurement locations $X_t = X$ for all t, so that $C_t = C$ and $W_t = W$. This assumption is standard in Kalman filter convergence theory [192, Chapter 4] and simplifies the notation, though the pointwise-in-time error identity derived below holds without

it.

The analytical challenge is that the filter operates on a finite-dimensional subspace while the true state lives in $L_2(\mathcal{X};\mathbb{R}^D)$. When f_t does not lie in $\mathrm{span}\{u_1,\dots,u_M\}$, the evolution kernel and the point measurements couple the modeled and unmodeled components, producing leakage in both the coefficient dynamics and the observations that the filter ignores. The filter covariance $\Psi_{t|t}$ consequently underestimates the true posterior uncertainty. This overconfidence, and the leakage from unresolved scales that drives it, are well documented in reduced-order and ensemble data assimilation, where they are handled through covariance inflation or consider-state corrections [193]. The contribution of this section is to characterize them exactly in the infinite-dimensional setting. We derive an exact decomposition of the functional estimation error that separates the noise-induced and truncation-induced contributions, and show that the approximation error vanishes as $M \to \infty$ at a rate set by the kernel's smoothness, so that the reduced posterior recovers the exact infinite-dimensional one.

7.5.1 Stability of the Estimation Loop

If the evolution kernel K_f is square-integrable, then $\mathcal{K}$ is a Hilbert–Schmidt operator and therefore compact [194]. Its spectrum consists of at most a countable set of eigenvalues with the only accumulation point at zero [195, Theorem 3.9].

Definition 7.7. The state dynamics (7.1) are called *stable* if the spectral radius of the integral operator satisfies $\rho(\mathcal{K}) := \sup\{|\lambda| : \lambda \in \sigma(\mathcal{K})\} < 1$.

Let $\pi_M : L_2(\mathcal{X}) \to \mathrm{span}\{u_1,\dots,u_M\}$ denote the orthogonal projection and $\mathcal{K}_M := \pi_M \mathcal{K} \pi_M$ the projected operator, whose matrix representation is A_M. The following convergence result is a classical consequence of Galerkin approximation theory for compact operators.

Proposition 7.8 (Galerkin convergence, [195, Chapters 10 and 13])**.** *If the basis $\{u_m\}_{m=1}^{\infty}$ is complete in $L_2(\mathcal{X})$, then $\|\mathcal{K} - \mathcal{K}_M\| \to 0$ as $M \to \infty$. In particular, $\rho(\mathcal{K}) < 1$ implies $\rho(A_M) < 1$ for all sufficiently large M.*

Compactness of $\mathcal{K}$ is what makes this convergence uniform rather than pointwise in f, so that the truncation error is controlled for every field the model can produce, and not for one field at a time. The completeness requirement is satisfied by standard choices such as the Fourier basis and piecewise-constant (bin) functions.

By standard Kalman filter theory [192, Section 4.4], under detectability of (A_M, C) and stabilizability of $(A_M, \Lambda_v^{1/2})$, the filter covariance $\Psi_{t|t}$ from Lemma 7.4 converges to a unique positive semi-definite limit $\Psi_\infty := \lim_{t\to\infty} \Psi_{t|t}$. When f_t lies exactly in $\mathrm{span}\{u_1,\dots,u_M\}$ for all t, the filter covariance $\Psi_{t|t}$ equals the true error

covariance $\mathbb{E}[e_t e_t^\top]$, and the filter is the exact MMSE estimator. Let Γ_∞ denote the corresponding steady-state Kalman gain and define

$$A_\Gamma := A_M - \Gamma_\infty C. \tag{7.32}$$

The closed-loop matrix A_Γ is Schur stable, that is, $\rho(A_\Gamma) < 1$. This holds regardless of $\rho(\mathcal{K})$. The DGP filter stably estimates functions with unstable dynamics, provided the system is detectable from the available measurement locations.

7.5.2 Steady-State Estimation Error for Finite M

In general, the function f_t does not lie exactly in $\operatorname{span}\{u_1, \dots, u_M\}$, so the basis approximation introduces an additional error on top of the noise-induced estimation error. To quantify this effect, let $\pi_M^\perp := I - \pi_M$ and note that the filter estimates only the in-subspace component $\hat{f}_{t|t}(x) = \check{U}^\top(x)\,\hat{z}_{t|t}$. The true pointwise error therefore decomposes as

$$f_t(x) - \hat{f}_{t|t}(x) = \check{U}^\top(x)\, e_t + (\pi_M^\perp f_t)(x), \tag{7.33}$$

where z_t is the coefficient representation of $\pi_M f_t$ as defined in Section 7.4.2, and $e_t := z_t - \hat{z}_{t|t} \in \mathbb{R}^{DM}$ denotes the in-subspace estimation error, i.e., the discrepancy between the true projection coefficients and the filter's estimate. The second term $(\pi_M^\perp f_t)(x)$ is the out-of-subspace residual that no finite-dimensional filter can correct.

Projecting the true dynamics $f_t = \mathcal{K} f_{t-1} + v_t$ onto the basis yields

$$z_t = A_M z_{t-1} + \pi_M \mathcal{K}\, \pi_M^\perp f_{t-1} + v_t^M, \tag{7.34}$$

with v_t^M of covariance Λ_v. The term $\pi_M \mathcal{K}\, \pi_M^\perp f_{t-1}$ feeds the unmodeled field $\pi_M^\perp f_{t-1}$ into the coefficients through the evolution kernel. The measurements do the same: evaluating the true function at the observation locations gives $Y_t = C\, z_t + \Phi\,(\pi_M^\perp f_t)(X) + w_t$, so the residual is observed together with the coefficients. The filter ignores both couplings. The filter update for the coefficient vector, from Lemma 7.6 with the steady-state gain Γ_∞, is

$$\hat{z}_{t|t} = \hat{z}_{t|t-1} + \Gamma_\infty(Y_t - C\,\hat{z}_{t|t-1}), \qquad \hat{z}_{t+1|t} = A_M\,\hat{z}_{t|t}. \tag{7.35}$$

Subtracting the filter update (7.35) from (7.34) gives the coefficient-error dynamics

$$e_t = A_\Gamma\, e_{t-1} + \ell_t - \Gamma_\infty w_t + v_t^M, \qquad \ell_t := \pi_M \mathcal{K}\, \pi_M^\perp f_{t-1} - \Gamma_\infty \Phi\,(\pi_M^\perp f_{t-1})(X). \tag{7.36}$$

Here ℓ_t is the leakage that the filter ignores, gathering the unmodeled field as it enters through the evolution and through the measurements. Both terms are

functions of $\pi_M^{\perp} f_{t-1}$, whose second moments may or may not converge depending on the state dynamics. The following two results formalize this.

Proposition 7.9. *If the state dynamics are stable in the sense of Definition 7.7 and the filter is stable ($\rho(A_\Gamma) < 1$), then f_t and e_t have bounded second moments, and the following limits exist:*

$$\Omega_M := \lim_{t\to\infty} \mathbb{E}[\ell_t \ell_t^\top] \succeq 0, \tag{7.37}$$

$$\Xi_M := \lim_{t\to\infty} \mathbb{E}[e_{t-1}\, \ell_t^\top]. \tag{7.38}$$

The cross-covariance Ξ_M is in general nonzero because e_{t-1} and ℓ_t both depend on f_{t-1}.

Proof. Stability of the state dynamics ($\rho(\mathcal{K}) < 1$) ensures that f_t converges to a stationary distribution. This distribution is the discrete-time counterpart of the invariant measure of a linear stochastic evolution equation with additive noise [196, Section 11.3]. Since ℓ_t is a bounded linear transformation of f_{t-1}, its second moment is bounded by a multiple of $\mathbb{E}[\|f_{t-1}\|^2]$, so Ω_M inherits the convergence of f_t. The error e_t is driven by ℓ_t, w_t, and v_t^M through the stable recursion (7.36), so its second moments also converge, and convergence of Ξ_M follows from the Cauchy–Schwarz inequality. □

Proposition 7.10. *Let (A_M, C) be detectable and $(A_M, \Lambda_v^{1/2})$ stabilizable, so that $\rho(A_\Gamma) < 1$, and suppose Ω_M and Ξ_M defined in (7.37)–(7.38) are finite. Then the true steady-state in-subspace error covariance $P_\infty := \lim_{t\to\infty} \mathbb{E}[e_t e_t^\top]$ satisfies the modified discrete Lyapunov equation*

$$P_\infty = A_\Gamma\, P_\infty\, A_\Gamma^\top + \Gamma_\infty W \Gamma_\infty^\top + \Lambda_v + \Omega_M + A_\Gamma\, \Xi_M + \Xi_M^\top A_\Gamma^\top. \tag{7.39}$$

The steady-state filter covariance Ψ_∞ satisfies

$$\Psi_\infty = A_\Gamma\, \Psi_\infty\, A_\Gamma^\top + \Gamma_\infty W \Gamma_\infty^\top + \Lambda_v, \tag{7.40}$$

which is (7.39) without the leakage terms Ω_M and Ξ_M.

Proof. Recall the error dynamics (7.36): $e_t = A_\Gamma\, e_{t-1} + \ell_t - \Gamma_\infty w_t + v_t^M$. Taking the steady-state second moment and expanding yields

$$P_\infty = A_\Gamma\, P_\infty\, A_\Gamma^\top + \mathbb{E}[\ell_t \ell_t^\top] + \Gamma_\infty W \Gamma_\infty^\top + \Lambda_v + A_\Gamma\, \mathbb{E}[e_{t-1}\, \ell_t^\top] + \mathbb{E}[\ell_t\, e_{t-1}^\top]\, A_\Gamma^\top,$$

where w_t and v_t^M are independent of (e_{t-1}, ℓ_t) because ℓ_t depends only on f_{t-1}. Identifying $\Omega_M = \mathbb{E}[\ell_t \ell_t^\top]$ and $\Xi_M = \mathbb{E}[e_{t-1}\, \ell_t^\top]$ from Proposition 7.9 gives (7.39).

The filter's Lyapunov equation (7.40) follows because the filter assumes $\ell_t \equiv 0$, which eliminates Ω_M and Ξ_M. Proposition 7.9 gives sufficient conditions for these to be finite, though Proposition 7.10 itself does not require $\rho(\mathcal{K}) < 1$. □

Corollary 7.11. *Suppose $\rho(\mathcal{K}) < 1$. The field f_t then has a unique stationary covariance operator Σ_f, given by the solution of the operator Lyapunov equation*

$$\Sigma_f = \mathcal{K}\,\Sigma_f\,\mathcal{K}^* + \mathcal{Q}_v, \tag{7.41}$$

with $\mathcal{Q}_v$ the covariance operator of v_t. The leakage terms Ω_M and Ξ_M of Proposition 7.10, and hence P_∞, are determined by Σ_f. In particular, P_∞ follows from the stationary covariance of the joint linear-Gaussian system formed by the filter and the basis coefficients of the field, computed by a finite-dimensional discrete Lyapunov equation and without Monte Carlo simulation.

Proof. By Proposition 7.10, P_∞ solves the modified Lyapunov equation (7.39), whose terms Ω_M and Ξ_M are second moments of the leakage ℓ_t, a bounded linear map of $\pi_M^\perp f_{t-1}$. For $\rho(\mathcal{K}) < 1$ the field is stationary and its covariance operator solves (7.41) [196, Section 11.3]. Expanding the field in the basis and stacking its coefficients with the filter estimate gives a joint linear-Gaussian system driven by white noise. Its stationary covariance solves a discrete Lyapunov equation, from which P_∞, the covariance of $e_t = z_t - \hat{z}_{t|t}$, is obtained by a linear map. □

The reduced filter reports Ψ_∞, whereas P_∞ from (7.39) is available from the model before any data are collected. Reporting P_∞ in place of Ψ_∞ removes the overconfidence exactly.

Remark 7.12. *The leakage ℓ_t has two parts: an evolution part $\pi_M \mathcal{K} \pi_M^\perp f_{t-1}$, and a measurement part from sampling $\pi_M^\perp f_{t-1}$ at the observation locations. When the basis functions are eigenfunctions of $\mathcal{K}$, the operator maps the resolved and unresolved subspaces separately, so $\pi_M \mathcal{K} \pi_M^\perp = 0$ and the evolution part vanishes. The Fourier basis does this for translation-invariant kernels [197]. Even then the approximation is inexact, because the disturbance excites the unresolved modes and the sensors sample them, so the measurement part and the out-of-subspace residual persist.*

The quantities P_∞ and Ψ_∞ characterize the in-subspace error e_t as a matrix in $\mathbb{R}^{DM \times DM}$. In practice, the quantity of interest is the functional estimation error $\|f_t - \hat{f}_{t|t}\|_{L_2}$. The following result connects the two.

Proposition 7.13. *For all $t \geq 0$,*

$$\mathbb{E}\Big[\|f_t - \hat{f}_{t|t}\|_{L_2}^2\Big] = \operatorname{tr}\big((I_D \otimes \Lambda_U)\,\mathbb{E}[e_t\, e_t^\top]\big) + \mathbb{E}\big[\|\pi_M^\perp f_t\|_{L_2}^2\big]\,, \tag{7.42}$$

with Λ_U the Gram matrix of the basis defined in (7.24). In particular, as $t \to \infty$ under the conditions of Proposition 7.10,

$$\lim_{t\to\infty} \mathbb{E}\Big[\|f_t - \hat{f}_{t|t}\|_{L_2}^2\Big] = \operatorname{tr}\big((I_D \otimes \Lambda_U)\,P_\infty\big) + \lim_{t\to\infty} \mathbb{E}\big[\|\pi_M^\perp f_t\|_{L_2}^2\big]\,. \tag{7.43}$$

Proof. By (7.33), the error decomposes as $f_t(x) - \hat{f}_{t|t}(x) = \check{U}^\top(x)\, e_t + (\pi_M^\perp f_t)(x)$. The first term lies in $\operatorname{span}\{u_1, \dots, u_M\}^D$ and the second in its orthogonal complement, so $\langle \check{U}^\top(\cdot)\, e_t,\, \pi_M^\perp f_t \rangle_{L_2} = 0$ for every realization. By the Pythagorean theorem,

$$\|f_t - \hat{f}_{t|t}\|_{L_2}^2 = \|\check{U}^\top(\cdot)\, e_t\|_{L_2}^2 + \|\pi_M^\perp f_t\|_{L_2}^2.$$

Computing the first term:

$$\begin{aligned}\|\check{U}^\top(\cdot)\, e_t\|_{L_2}^2 &= e_t^\top \left(\int_{\mathcal{X}} \check{U}(x)\, \check{U}^\top(x)\, d\nu(x) \right) e_t \\ &= e_t^\top (I_D \otimes \Lambda_U)\, e_t.\end{aligned}$$

Taking expectations and using $\mathbb{E}[e_t^\top A\, e_t] = \operatorname{tr}(A\, \mathbb{E}[e_t\, e_t^\top])$ gives (7.42). □

The first term in (7.42) is the in-subspace estimation error, weighted by the Gram matrix of the basis functions. The second term is the out-of-subspace residual that no finite-dimensional filter can correct. For an orthonormal basis ($\Lambda_U = I_M$), the first term reduces to $\operatorname{tr}(\mathbb{E}[e_t\, e_t^\top])$.

The filter reports $\Psi_{t|t}$ as its error covariance, but this differs from $\mathbb{E}[e_t\, e_t^\top]$ when the basis projection is inexact. The following proposition quantifies the gap between P_∞ and Ψ_∞.

Proposition 7.14. *Let (A_M, C) be detectable and $(A_M, \Lambda_v^{1/2})$ stabilizable, and suppose Ω_M and Ξ_M exist. Then $P_\infty - \Psi_\infty$ satisfies the discrete Lyapunov equation*

$$P_\infty - \Psi_\infty = A_\Gamma (P_\infty - \Psi_\infty) A_\Gamma^\top + \Omega_M + A_\Gamma\, \Xi_M + \Xi_M^\top A_\Gamma^\top. \tag{7.44}$$

In particular, $P_\infty \succeq \Psi_\infty$ whenever $\Omega_M + A_\Gamma\, \Xi_M + \Xi_M^\top A_\Gamma^\top \succeq 0$, and the gap satisfies

$$\|P_\infty - \Psi_\infty\| \le \frac{c_\Gamma^2\, \|\Omega_M + A_\Gamma\, \Xi_M + \Xi_M^\top A_\Gamma^\top\|}{1 - \beta^2}, \tag{7.45}$$

in which $\beta \in (\rho(A_\Gamma), 1)$ and $c_\Gamma \ge 1$ are constants for which $\|A_\Gamma^k\| \le c_\Gamma\, \beta^k$ holds for all k.

Proof. Subtracting the Lyapunov equations for Ψ_∞ and P_∞ gives (7.44). Iterating yields the explicit solution $P_\infty - \Psi_\infty = \sum_{k=0}^{\infty} A_\Gamma^k\, (\Omega_M + A_\Gamma\, \Xi_M + \Xi_M^\top A_\Gamma^\top)\, (A_\Gamma^\top)^k$. If the driving term is positive semidefinite, every summand is positive semidefinite, so $P_\infty \succeq \Psi_\infty$. Since A_Γ is Schur stable, constants c_Γ and β with the stated property exist, and taking norms and summing the geometric series gives the bound. □

This result generalizes the reduced-order Kalman filter analysis of [191] and [177] to the setting where the full model is infinite-dimensional rather than a finite-dimensional truncation. An optimal reduced-order filter in the sense of [198] would additionally account for Ω_M and Ξ_M in the gain computation. In the exact DGP case, $\pi_M^\perp f_t \equiv 0$ for all t, so $\Omega_M = \Xi_M = 0$ and $P_\infty = \Psi_\infty$.

Combining Propositions 7.13 and 7.14 gives a three-way decomposition of the steady-state functional error:

$$\lim_{t\to\infty} \mathbb{E}\Big[\|f_t - \hat{f}_{t|t}\|_{L_2}^2\Big] = \underbrace{\operatorname{tr}\big((I_D \otimes \Lambda_U)\,\Psi_\infty\big)}_{\text{noise-limited error}} + \underbrace{\operatorname{tr}\big((I_D \otimes \Lambda_U)(P_\infty - \Psi_\infty)\big)}_{\text{leakage-induced gap}} + \underbrace{\lim_{t\to\infty} \mathbb{E}\big[\|\pi_M^\perp f_t\|_{L_2}^2\big]}_{\text{out-of-subspace residual}}. \tag{7.46}$$

The first term can be computed directly from the filter output. The second is the in-subspace error from the filter ignoring the leakage ℓ_t, nonnegative whenever the condition in Proposition 7.14 holds, and the third is always nonnegative.

7.5.3 Convergence as $M \to \infty$

As the number of basis functions grows, all sources of approximation error vanish, and the approximate DGP recovers the exact infinite-dimensional estimator.

Proposition 7.15. *Under the conditions of Propositions 7.8 and 7.14, and assuming that the basis $\{u_m\}_{m=1}^\infty$ is complete in $L_2(\mathcal{X})$, the following limits hold as $M \to \infty$:*

1. *$\Omega_M \to 0$, $\Xi_M \to 0$, and $\|P_\infty - \Psi_\infty\| \to 0$, so that the overconfidence vanishes.*
2. *$\|\pi_M^\perp f_t\|_{L_2} \to 0$ for all t, so that the out-of-subspace residual vanishes.*
3. *The filter covariance Ψ_∞ converges to the steady-state posterior covariance of the exact infinite-dimensional DGP.*
4. *Hence, by items 1–3, the steady-state functional error (7.46) converges to that of the exact infinite-dimensional DGP.*

Proof. The first limit follows from Proposition 7.8 and completeness of the basis: both parts of the leakage vanish, since $\|\pi_M \mathcal{K} \pi_M^\perp\| \to 0$ and $\|\pi_M^\perp f_t\|_{L_2} \to 0$, so that $\Omega_M \to 0$ and $\Xi_M \to 0$. Combined with the bound in (7.45), this gives $\|P_\infty - \Psi_\infty\| \to 0$. The second limit follows from completeness of the basis. The third limit follows from the convergence of Galerkin approximations of operator Riccati equations [199, 200]. The fourth limit follows from Proposition 7.13 and items 1–3. □

Convergence of such finite-dimensional approximations to the optimal infinite-dimensional filter is classical [199, 200]. The rate below instead governs the overconfidence of the reduced-rank filter, and is set by how fast the field can be represented in the basis.

Proposition 7.16. *Let* $r_M := \lim_{t\to\infty} \mathbb{E}[\|\pi_M^{\perp} f_t\|_{L_2}^2]$ *be the steady-state out-of-subspace variance. Under the conditions of Proposition 7.14 and for a uniformly bounded basis, there is a constant* c *independent of* M *with*

$$\|P_\infty - \Psi_\infty\| \le c\sqrt{r_M}. \tag{7.47}$$

The overconfidence gap and the out-of-subspace residual therefore vanish at the rates $\sqrt{r_M}$ *and* r_M*.*

Proof. The leakage ℓ_t is a bounded linear map of $\pi_M^{\perp} f_{t-1}$, so $\|\Omega_M\| \le \operatorname{tr}\Omega_M \le c_1 r_M$, with c_1 finite for a uniformly bounded basis. The recursion (7.36) is stable and driven by inputs whose second moments are bounded uniformly in M, so $\operatorname{tr} P_\infty$ is bounded uniformly as well, and the Cauchy–Schwarz inequality gives $\|\Xi_M\| \le (\operatorname{tr} P_\infty \operatorname{tr}\Omega_M)^{1/2} \le c_2\sqrt{r_M}$. The driving term of (7.44) then satisfies $\|\Omega_M + A_\Gamma \Xi_M + \Xi_M^\top A_\Gamma^\top\| \le c_1 r_M + 2\|A_\Gamma\| c_2\sqrt{r_M}$, and (7.45) yields (7.47). □

The rate is governed by the decay of $r_M = \sum_{n>M} \sigma_n$, the tail of the field's modal variances $\sigma_n := \langle u_n, \Sigma_f u_n\rangle$. Smoother kernels and disturbances give faster-decaying modal variances, and hence a faster-vanishing tail: r_M decays geometrically when they are analytic, and as $M^{-2s/d}$ under Sobolev smoothness s on a d-dimensional domain [197]. Section 7.6.1 illustrates the analytic case.

7.6 Numerical Examples

This section presents two numerical examples that demonstrate the DGP estimator and its separable kernel approximation from Section 7.4, as well as the approximation-induced error from Section 7.5. The heat equation example illustrates the approximation error and its convergence as M grows. The wave equation conserves energy and is only marginally stable, so it demonstrates the vector-valued extension to a second-order-in-time PDE rather than the steady-state analysis. Each example begins with the derivation of the evolution kernel from the underlying PDE, following the connection described in Section 7.2.1.

7.6.1 Example 1: Heat Equation

Consider the one-dimensional heat equation on $\mathcal{X} = \mathbb{R}$,

$$\frac{\partial f}{\partial \tau}(x,\tau) = \alpha \frac{\partial^2 f}{\partial x^2}(x,\tau), \tag{7.48}$$

in which $\alpha > 0$ is the thermal diffusivity. The Green's function of (7.48) on $\mathbb{R}$ is [190, Section 2.3.1]

$$G(x, s, \tau) = \frac{1}{\sqrt{4\pi\alpha\tau}} \exp\left(-\frac{(x-s)^2}{4\alpha\tau}\right), \tag{7.49}$$

which maps the initial condition at location s to the solution at location x after time τ.

Evolution Kernel

Defining $f_t(x) := f(x, t\Delta)$ for $t \in \mathbb{N}$ and setting $\tau = \Delta$ in (7.49) gives the evolution kernel

$$k_f(x, s) = \frac{1}{\sqrt{4\pi\alpha\Delta}} \exp\left(-\frac{(x-s)^2}{4\alpha\Delta}\right), \tag{7.50}$$

which is a squared exponential kernel with length scale $\sigma_k = \sqrt{2\alpha\Delta}$ and amplitude $a_k = (2\pi\sigma_k^2)^{-1/2}$. This is a scalar ($D = 1$) problem, so K_f reduces to a scalar-valued kernel.

Setup

Consider the heat equation on the bounded domain $\mathcal{X} = [-1, 1]$ with evolution kernel (7.50). On a bounded domain, the system is naturally strictly stable in the sense of Definition 7.7, since heat that diffuses past the domain boundary is lost.

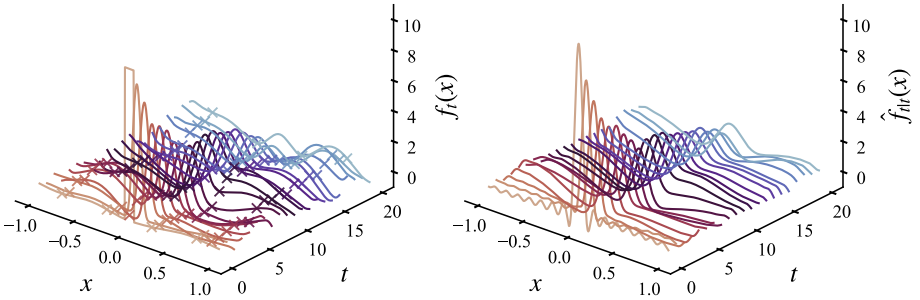
To assess the estimation accuracy, a ground truth model is generated using 625 discretized basis functions, while the approximate DGP uses up to 50 Fourier bases. The initial condition covariance, process noise covariance, and measurement noise covariance are

$$Q_f(x, x') = a_f \exp\left(-\frac{(x-x')^2}{2\sigma_f^2}\right), \qquad Q_v(x, x') = a_v \exp\left(-\frac{(x-x')^2}{2\sigma_v^2}\right),$$
$$Q_w(x, x') = \sigma_w^2\, \delta(x - x'),$$

with $a_f = 0.1$, $\sigma_f = 0.3$, $a_v = 0.1$, $\sigma_v^2 = 0.1$, and $\sigma_w^2 = 0.1$. The initial condition covariance Q_f ensures that f_0 smoothly deviates from its mean. The measurement noise Q_w is spatially white. The initial mean function is

$$\bar{f}_0(x) = \begin{cases} 10 & |x| < 0.05, \\ 0 & \text{otherwise}, \end{cases}$$

which simulates an impulse-like function at the center of the domain. Observations are taken at $p = 4$ fixed locations, spread evenly over $\mathcal{X}$. Errors are reported in the Euclidean norm on the simulation grid, which is proportional to the L_2 norm.



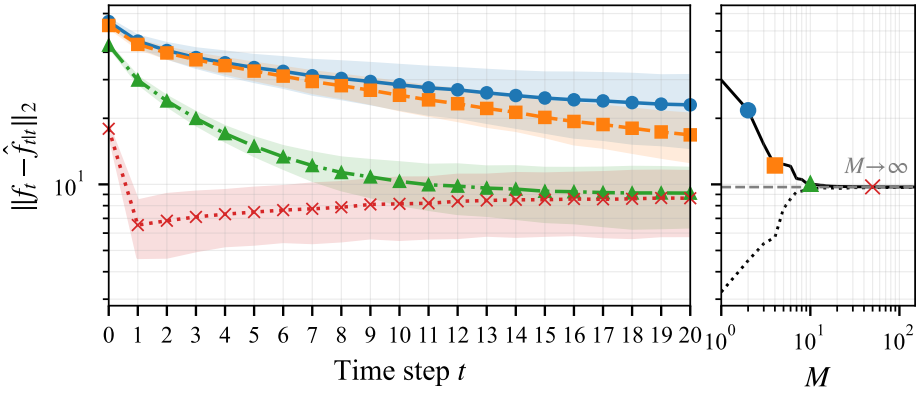
(a) Ground truth **(b)** DGP estimate

Figure 7.1. Heat equation example. The ground truth $f_t(x)$ in **(a)** is shown with the fixed observation locations (✖), and the DGP estimate $\hat{f}_{t|t}(x)$ in **(b)** uses $M = 31$ Fourier basis functions. Each line represents one discrete time step; line color indicates the time index.

7

Figure 7.1 shows the ground truth and the DGP estimate side by side. The smoothing of the function as the time index t increases is characteristic of the heat equation: the initial impulse diffuses and spreads over the spatial domain. The effect of the process disturbances v_t is visible in the ground truth, for instance near $x = -0.5$ at time step $t = 3$. The DGP estimate tracks the evolving function well despite these disturbances. At $t = 0$, however, a noticeable approximation error is visible: the discontinuous initial condition $\bar{f}_0$ cannot be exactly represented by a finite number of smooth Fourier basis functions, and the resulting overshoot is a manifestation of the Gibbs phenomenon. Because the heat kernel smooths the function over time, this representation error vanishes quickly as the ground truth becomes increasingly well-suited to the Fourier basis.

Figure 7.2 shows the 2-norm of the estimation error for different numbers of basis functions M. Increasing M reduces the estimation error with diminishing returns, which is consistent with Proposition 7.15: as M increases, the leakage-induced gap and the out-of-subspace residual in (7.46) decrease toward zero, leaving only the irreducible noise-limited error. The heat kernel (7.50) is a squared-exponential and therefore analytic, so its modal variances decay geometrically. By Proposition 7.16 the approximation error then decays geometrically as well, the fast, diminishing-returns convergence seen here. The error also decreases over time for all choices of M, which can be explained by the smoothing effect of the heat kernel: as the true function evolves, it becomes easier to represent in the Fourier basis. This effect is least visible for $M = 50$, which can already represent non-smooth functions such as the initial impulse. The steady-state error in Figure 7.2**(b)** shows the filter's reported error lying below the actual error, the overconfidence of Proposition 7.14, with the gap closing as M grows. The exact error covariance of Corollary 7.11



(a) True estimation error **(b)** $t \to \infty$ error

Figure 7.2. Convergence for the heat equation example. The 2-norm of the estimation error $\|f_t - \hat{f}_{t|t}\|_2$ over time steps t is shown in **(a)**, for $M = 2$ (-●-), $M = 4$ (-■-), $M = 10$ (-▲-), and $M = 50$ (-×-) Fourier bases. Lines show the mean over 500 Monte Carlo runs. Shaded regions in the matching color indicate ± 1 standard deviation. The root-mean-square steady-state estimation error $\left(\mathbb{E}[\|f_\infty - \hat{f}_{\infty|\infty}\|_2^2]\right)^{1/2}$ in **(b)** is given as a function of M (—), with markers at $M \in \{2, 4, 10, 50\}$ using the same colors and markers throughout. The dotted black line (····) is the filter's reported error, the square root of the noise-limited term of (7.46). It lies below the actual error, confirming the overconfidence of Proposition 7.14, and the gap closes as M grows. The dashed gray line (- - -), labeled $M \to \infty$, indicates the full-order limit.

corrects this optimistic value.

7.6.2 Example 2: Wave Equation

Consider the one-dimensional wave equation on $\mathcal{X} = \mathbb{R}$,

$$\frac{\partial^2 \varphi}{\partial \tau^2}(x, \tau) = c^2 \frac{\partial^2 \varphi}{\partial x^2}(x, \tau), \tag{7.51}$$

in which $c > 0$ is the wave propagation speed. The wave equation is second-order in time, and its solution depends on two initial conditions: the initial position $\varphi(x, 0) = \varphi_0(x)$ and the initial velocity $\psi(x, 0) := \frac{\partial \varphi}{\partial \tau}(x, 0) = \psi_0(x)$. By D'Alembert's formula, the solution is

$$\varphi(x, \tau) = \frac{1}{2}[\varphi_0(x - c\tau) + \varphi_0(x + c\tau)] + \frac{1}{2c} \int_{x-c\tau}^{x+c\tau} \psi_0(s)\, ds. \tag{7.52}$$

Evolution Kernels

Because the wave equation is second-order in time, we define $\varphi_t(x) := \varphi(x, t\Delta)$ and $\psi_t(x) := \frac{\partial \varphi}{\partial \tau}(x, t\Delta)$ for $t \in \mathbb{N}$, and set $f_t(x) := [\varphi_t(x),\, \psi_t(x)]^\top \in \mathbb{R}^2$. Substituting the state at discrete time t as the initial condition and evaluating D'Alembert's formula (7.52) at $\tau = \Delta$ gives the position update

$$\varphi_{t+1}(x) = \frac{1}{2}[\varphi_t(x - c\Delta) + \varphi_t(x + c\Delta)] + \frac{1}{2c}\int_{x-c\Delta}^{x+c\Delta} \psi_t(s)\, ds. \tag{7.53}$$

The velocity update is obtained by differentiating (7.52) with respect to time. By the chain rule, $\frac{\partial}{\partial \tau}\varphi_t(x \pm c\tau)\big|_{\tau=\Delta}$ produces a factor $\pm c$ and converts the time derivative into a spatial derivative φ_t', giving

$$\psi_{t+1}(x) = \frac{c}{2}[\varphi_t'(x + c\Delta) - \varphi_t'(x - c\Delta)] + \frac{1}{2}[\psi_t(x - c\Delta) + \psi_t(x + c\Delta)]\,, \tag{7.54}$$

where $\varphi_t'(\cdot) := \frac{d}{d(\cdot)}\varphi_t(\cdot)$ denotes the spatial derivative.

Following (7.5), the evolution kernels are identified by rewriting (7.53)–(7.54) as integrals against the state. From (7.53), the shifted evaluations of φ_t and the integral over ψ_t give

$$\begin{aligned} k_{\varphi\varphi}(x, s) &= \tfrac{1}{2}[\delta(x - s - c\Delta) + \delta(x - s + c\Delta)]\,, \\ k_{\varphi\psi}(x, s) &= \tfrac{1}{2c}\,\mathbf{1}_{[x-c\Delta,\, x+c\Delta]}(s), \end{aligned} \tag{7.55}$$

where $\mathbf{1}_{[a,b]}(s)$ denotes the indicator function on $[a, b]$. From (7.54), the spatial derivatives of φ_t yield

$$\begin{aligned} k_{\psi\varphi}(x, s) &= \tfrac{c}{2}[\delta'(x - s + c\Delta) - \delta'(x - s - c\Delta)]\,, \\ k_{\psi\psi}(x, s) &= k_{\varphi\varphi}(x, s), \end{aligned} \tag{7.56}$$

where δ' denotes the distributional derivative of the Dirac delta. The symmetry $k_{\psi\psi} = k_{\varphi\varphi}$ is a direct consequence of D'Alembert's formula. With these kernels, the updates (7.53)–(7.54) take the form of a coupled IDE:

$$\underbrace{\begin{bmatrix}\varphi_{t+1}(x)\\ \psi_{t+1}(x)\end{bmatrix}}_{f_{t+1}(x)} = \int_{\mathcal{X}} \underbrace{\begin{bmatrix}k_{\varphi\varphi}(x,s) & k_{\varphi\psi}(x,s)\\ k_{\psi\varphi}(x,s) & k_{\psi\psi}(x,s)\end{bmatrix}}_{K_f(x,s)\,\in\,\mathbb{R}^{2\times 2}} \underbrace{\begin{bmatrix}\varphi_t(s)\\ \psi_t(s)\end{bmatrix}}_{f_t(s)} ds, \tag{7.57}$$

which is an instance of (7.1) with $D = 2$. Because the kernels in (7.55)–(7.56) are distributions rather than ordinary functions, they are replaced by Gaussian approximations with smoothing parameter $\epsilon > 0$ for the numerical implementation. The wave equation conserves energy, so $\rho(\mathcal{K}) = 1$ and the steady-state results of

Propositions 7.9–7.15 do not formally apply. However, the closed-loop matrix A_Γ from (7.32) is Schur stable under detectability of (A_M, C), whether or not the open-loop system is stable, so the filter can track a bounded signal even when the open-loop dynamics are marginally stable. In the disturbance-free setting below, the ground truth is the bounded analytical solution, and the estimation error remains bounded because the filter stably incorporates new observations at each time step.

Setup

The example is evaluated on the bounded domain $x \in [-10, 10]$ with wave speed $c = 2$ and time step $\Delta = 0.2$. The state is $f_t(x) = [\varphi_t(x),\, \psi_t(x)]^\top$, where φ_t is the position function and ψ_t is the velocity function. The distributional evolution kernels are replaced by Gaussian approximations with smoothing parameter $\epsilon = h$, where $h > 0$ is the spatial grid spacing. As $\epsilon \to 0$, these smoothed kernels converge to the exact distributional kernels.

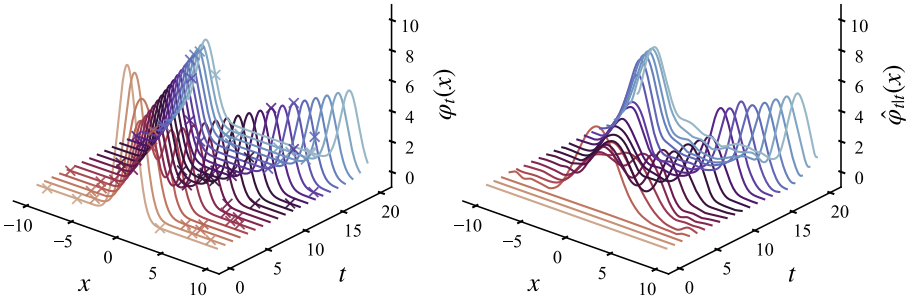
The ground truth is the analytical D'Alembert solution $\varphi(x, \tau) = \frac{1}{2}[\varphi_0(x - c\tau) + \varphi_0(x + c\tau)]$ with initial position $\varphi_0(x) = 10\exp(-x^2/2)$ and initial velocity $\psi_0(x) = 0$. This produces two counter-propagating Gaussian pulses. Because this solution is deterministic, there are no process disturbances. The DGP estimator uses $M = 31$ Fourier basis functions per state component (so $2M = 62$ coefficients in total) with $p = 3$ randomly placed measurements of φ_t at each time step and measurement noise variance $\sigma_w^2 = 10^{-5}$. The prior mean is set to zero, so that the estimator must discover the propagating wavefronts entirely from the observations.

Unlike the heat equation, which diffuses energy over the domain, the wave equation preserves the shape of propagating wavefronts. This makes the estimation problem fundamentally different: the evolution kernel is not smoothing, and the function does not become easier to approximate over time.

Figure 7.3 shows the ground truth and the DGP estimate for the wave equation. The two counter-propagating pulses are clearly visible in the analytical solution. Despite starting from a zero prior mean and receiving only $p = 3$ noisy measurements per time step, the DGP estimator recovers the shape of both propagating wavefronts within a few time steps. Because the wave equation does not smooth the solution, the Fourier basis must resolve the sharp features of the pulse at every time step, in contrast to the heat equation example where the kernel itself reduces the required bandwidth over time.

7.7 Conclusions

This chapter presented the Dynamic Gaussian Process (DGP), which unifies Gaussian process regression and Kalman filtering for the estimation of functions governed



(a) Ground truth **(b)** DGP estimate

Figure 7.3. Wave equation example. The analytical D'Alembert solution $\varphi_t(x)$ in **(a)** is shown with observation locations (✖), and the DGP estimate $\hat{\varphi}_{t|t}(x)$ in **(b)** uses $M = 31$ Fourier bases. Each line represents one discrete time step; line color indicates the time index.

by integro-difference equations. The stability and approximation error analysis showed that the functional estimation error decomposes exactly into three distinct components. Specifically, this error comprises the noise-limited error, the leakage-induced gap, and the out-of-subspace residual. The leakage-induced gap and the out-of-subspace residual vanish as $M \to \infty$ at a rate set by the smoothness of the kernel, while the noise-limited error converges to the irreducible posterior uncertainty of the exact infinite-dimensional DGP. All three terms are computable from the model, so the reduced filter's overconfident covariance can be corrected to the true error without sampling. The heat equation example confirms this convergence, while the wave equation example demonstrates estimation of a PDE that is second-order in time, handled through the vector-valued state.

Several directions for future research remain. Extensions to generalize the model with control inputs or to include nonlinear PDEs are of interest. The posterior covariance of the DGP can also be used to determine where to place the next observation, enabling frameworks for sensor scheduling and Bayesian optimization.

8

Data-Efficient Quadratic Q-Learning Using LMIs

Abstract - Reinforcement learning (RL) has seen significant research and application results but often requires large amounts of training data. This chapter proposes two data-efficient off-policy RL methods that use parametrized Q-learning. In these methods, the Q-function is chosen to be linear in the parameters and quadratic in selected basis functions in the state and control deviations from a base policy. A cost penalizing the ℓ_1-norm of Bellman errors is minimized. We propose two methods: Linear Matrix Inequality Q-Learning (LMI-QL) and its iterative variant (LMI-QLi), which solve the resulting episodic optimization problem through convex optimization. LMI-QL relies on a convex relaxation that yields a semidefinite programming (SDP) problem with linear matrix inequalities (LMIs). LMI-QLi entails solving sequential iterations of an SDP problem. Both methods combine convex optimization with direct Q-function learning to significantly improve learning speed. A numerical case study demonstrates their advantages over least-squares policy iteration, a standard parametrized Q-learning method. Code is available at `https://github.com/JvHulst/LMI_Q-Learning`.

This chapter is based on van Hulst et al. [201].

8.1 Introduction

Reinforcement learning (RL) has recently received significant attention due to excellent results in many application domains [202]. The data-driven nature of RL methods enables learning of high-performing policies, even without explicit models.

Q-learning, a common RL method, estimates the so-called quality of an action-state pair, representing the expected cumulative reward of choosing an action given a state. This quality learning uses the temporal difference error to update the Q-values iteratively and can be performed online or batch-wise. Q-learning in its most basic form tends to have slow convergence to the optimal policy, due to having to visit every state and action multiple times. Besides, it often requires the discretization of uncountable state and control spaces, which scales poorly as the dimension of the state space increases. To improve convergence speed and circumvent the curse of dimensionality, one can parameterize the Q-function using basis functions [203–205].

This chapter aims to accelerate Q-function convergence through convex optimization. Our approach consists of two key assumptions. First, the Q-function is chosen to be linear in the parameters and quadratic in a set of selected basis functions in the state, as well as quadratic in the control deviations from a given base policy, as detailed in Section 8.3. Second, we find the Q-function parameters by minimizing the ℓ_1-norm of the data-based Bellman errors. We propose two numerical methods to tackle the resulting episodic parameter optimization problem by solving convex subproblems. The first method reformulates the Bellman error minimization problem as a semidefinite programming problem by fixing the optimization variables that appear nonlinearly. We can then solve the resulting convex problem, update the previously fixed variables, and repeat the process iteratively. The second numerical method that we propose relies on a *convex relaxation* of the temporal difference minimization problem in the form of a semidefinite program including a set of linear matrix inequalities (LMIs). *Convex relaxations* provide efficient suboptimal solutions for difficult non-convex optimization problems while bounding the original cost function.

Even though both methods rely on convex optimization, the resulting solutions typically differ from the optimal Q-function due to the approximations introduced by coordinate descent and convex relaxations. Nonetheless, practical results demonstrate the effectiveness of the proposed approaches. Importantly, the proposed methods belong to the class of off-policy RL methods, and thus the data used can be generated in any way. We coin the term linear matrix inequality Q-learning (LMI-QL) for the convex relaxation method and linear matrix inequality Q-learning with iterations (LMI-QLi) for the iterative method.

Our proposed methods share similarities with existing approaches like least-squares temporal difference (LSTD) learning [206], which uses convex least-squares

optimization to improve data efficiency. Extensions such as recurrent least-squares temporal difference (RLSTD) offer further improvements. Least squares policy iteration (LSPI) [207] extends the idea of LSTD to the Q-function. In Lagoudakis [208], an overview of more least-squares RL methods is presented. Among these, LSPI performs approximate policy iteration, alternating policy evaluation with greedy policy improvement. Our methods instead target the optimal Q-function directly by minimizing the Bellman residual, rather than alternating evaluation and improvement. Like our methods, fitted Q-iteration learns the Q-function from a fixed batch of data [209], including the data-efficient neural variant of Riedmiller [210]. The convex structure we add lets each fit attain its global optimum.

Well-known RL methods like DQN [211], PPO [212], and MPO [213] achieve strong performance through various other innovations. DQN employs deep networks, which are extremely flexible but yield slow convergence in large spaces. PPO and MPO focus on stabilizing parameter updates rather than directly improving convergence speed. In contrast, our methods use structured parametrizations and convex optimization to achieve fast learning, particularly in problems where flexibility is less critical.

Although various results in the literature accelerate Q-learning using parametrization structures, to the best of the authors' knowledge, none combine the following two elements

1. direct learning of the optimal Q-function, rather than evaluating and improving a sequence of policies;

2. convex optimization, so that every optimization problem we solve attains its global optimum.

LMI-QL realizes the first element through a single convex program, whereas LMI-QLi solves a short sequence of convex programs. By combining direct Q-function learning with convex optimization, LMI-QL(i) learns accurate policies from little data. A numerical case study demonstrates both methods' effectiveness, comparing LMI-QL(i) to LSPI.

This chapter is organized as follows. Section 8.2 presents the problem tackled in this chapter, as well as provides some preliminaries on parametric Q-function learning. In Section 8.3, the proposed convex Q-learning methods are presented. In Section 8.4, we present the results of applying the methods to a nonlinear pendulum system. Lastly, Section 8.5 gives conclusions and recommendations for future research.

Notation. Let $\mathbb{R} = (-\infty, \infty)$. Let $\mathbb{N} = \{0, 1, \ldots\}$ denote the natural numbers. The i-th element of a vector x is denoted $x[i]$. The identity matrix of size n is denoted by I_n. A normal distribution with mean vector $\mu \in \mathbb{R}^n$ and covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$ is denoted $\mathcal{N}(\mu, \Sigma)$.

8.2 Problem Formulation

The notation in this section is largely taken from [202]. We consider a generic infinite-horizon Markov decision process (MDP) with stochastic state transitions. We denote the system state at discrete-time index $k \in \mathbb{N}$ by $x_k \in \mathbb{R}^{n_x}$, and the system input (or action) by $u_k \in \mathbb{R}^{n_u}$. The conditioned probability distribution that defines the state transition is fully characterized by $P : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to [0, \infty)$ where $\text{Prob}\left[x_{k+1} \in A \mid x_k = x, u_k = u\right] = \int_A P(x', x, u)dx'$ for $A \subseteq \mathbb{R}^{n_x}$, with $\int_{\mathbb{R}^{n_x}} P(x', x, u)dx' = 1$, for every $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$. The reward $r_k \in \mathbb{R}$ immediately following the state and input at time index $k \in \mathbb{N}$ is defined as

$$r_k = g(x_k, u_k) \tag{8.1}$$

with $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ the reward function.

The system input is often chosen as a function of the system state according to a so-called policy. We define a deterministic, time-invariant policy $\pi : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$ as

$$u_k = \pi(x_k), \quad k \in \mathbb{N} \tag{8.2}$$

A trajectory of states and inputs can be constructed, given the state transition dynamics characterized by P and the policy (8.2). From such a trajectory, we can obtain the infinite-horizon discounted return, which is given by

$$G_k = \sum_{i=0}^{\infty} \gamma^i r_{k+i} \tag{8.3}$$

with $\gamma \in (0, 1)$ the discount factor. The central problem in RL is to find a policy π^* that maximizes the expected value of the return. This expected value of the return under a policy π starting at state x_k is referred to as the value function of that policy, i.e.,

$$V^\pi(x_k) := \mathbb{E}\left[G_k \mid x_k, \pi\right]. \tag{8.4}$$

A related notion is the quality function or Q-function, which is defined as

$$Q^\pi(x_k, u_k) := \mathbb{E}\left[g(x_k, u_k) + \gamma V^\pi(x_{k+1}) \mid x_k, u_k, \pi\right]. \tag{8.5}$$

The quality function can play a key role in finding the optimal policy π^*, as

$$\pi^*(x_k) \in \arg\max_u Q^*(x_k, u), \tag{8.6}$$

in which Q^* is the optimal Q-function, defined by

$$Q^*(x_k, u) = \max_\pi Q^\pi(x_k, u). \tag{8.7}$$

In this sense, once the optimal Q-function Q^* is known, the optimal policy π^* follows directly via (8.6).

In this chapter, we make use of a parametrized quality function $\tilde{Q}(x_k, u_k, \theta)$ with $\theta \in \mathbb{R}^{n_\theta}$ a vector of parameters. The problem formulation is, given a batch of data $\mathcal{D} := \{\mathcal{X}, \mathcal{U}, \mathcal{R}, \mathcal{X}_+\}$,

$$\begin{aligned} \mathcal{X} &:= \{x_0,\ x_1,\ \ldots,\ \ x_N\}, \\ \mathcal{U} &:= \{u_0,\ u_1,\ \ldots,\ \ u_N\}, \\ \mathcal{R} &:= \{r_0,\ r_1,\ \ldots,\ \ r_N\}, \quad \text{and} \\ \mathcal{X}_+ &:= \{x_1,\ x_2,\ \ldots,\ x_{N+1}\}, \end{aligned}$$

find the parameters θ that yield Q-function $\tilde{Q}$, which best fits the data (in an approximated sense) according to the definition of the optimal quality function (8.7). In this sense, we are minimizing the "distance" between the parametrized Q-function $\tilde{Q}$ and the optimal Q-function Q^*, given the data. Note that the dataset $\mathcal{D}$ should be sufficiently informative, i.e., persistently exciting (PE) the parametrized quality function. We will later consider how this is achieved for the specific quality function structure proposed in this chapter.

8.3 Proposed Method

In this section, the proposed method is presented. We start by formalizing the problem setup by presenting the proposed cost and Q-function parametrization. Note that the cost discussed in this section refers to the objective function associated with the Q-function parameter optimization problem, rather than the MDP reward function. Afterward, we detail two proposed numerical methods.

8.3.1 Formalization of the Problem Setup

We propose to capture the objective of the previous section in the optimization problem

$$\min_{\theta} \|z\|_p \tag{8.8}$$

with $z := \left[z_0, z_1, \ldots, z_N\right]^\top$, in which

$$z_k := \tilde{Q}(x_k, u_k, \theta) - \left(r_k + \gamma \max_u \tilde{Q}(x_{k+1}, u, \theta)\right),$$

for $k \in \{0, 1, \ldots, N\}$. Here, $\|x\|_p$ represents the ℓ_p-norm of x. The quantity z_k is often referred to as the temporal difference or Bellman residual. This optimization problem captures the central problem of parametric Q-learning since the Bellman

residuals are zero for the optimal Q-function. The size of the residual also relates to the distance to Q^*. For the exact Bellman residual, if $\sup_{x,u}|\tilde{Q}(x,u)-(\mathcal{T}\tilde{Q})(x,u)| \leq \epsilon$ with $\mathcal{T}$ the Bellman optimality operator, then $\|\tilde{Q}-Q^*\|_\infty \leq \epsilon/(1-\gamma)$ [214]. This bound is idealized here, since it assumes a uniform bound over the whole state-action space, while we minimize the residual only on the finite sample $\mathcal{D}$. Minimizing the residual directly, as we do here, also avoids the divergence that semi-gradient temporal-difference methods can exhibit under off-policy data and function approximation [202, 215]. Note that in LSPI [207], the ℓ_2-version of this problem is considered. In this chapter, we will use the ℓ_1-norm ($p=1$) or absolute value norm. This particular choice is made since an absolute value norm cost admits a (convex) semidefinite programming form, as we will see.

It is assumed we have access to a baseline policy along with an approximation of its value function. The baseline policy is denoted by $\bar{\pi}:\mathbb{R}^{n_x}\to\mathbb{R}^{n_u}$, while the corresponding value function is denoted by $V^{\bar{\pi}}:\mathbb{R}^{n_x}\to\mathbb{R}$. We intend to learn the deviation from this baseline, i.e., $\Delta:\mathbb{R}^{n_x}\to\mathbb{R}^{n_u}$ in

$$u_k = \pi(x_k) = \bar{\pi}(x_k) + \Delta(x_k). \tag{8.9}$$

If no base policy is available, we can assume $\bar{\pi} = V^{\bar{\pi}} = 0$ and model the full Q-function with the parametrization that follows.

Given the policy baseline, we propose to parametrize the Q-function as

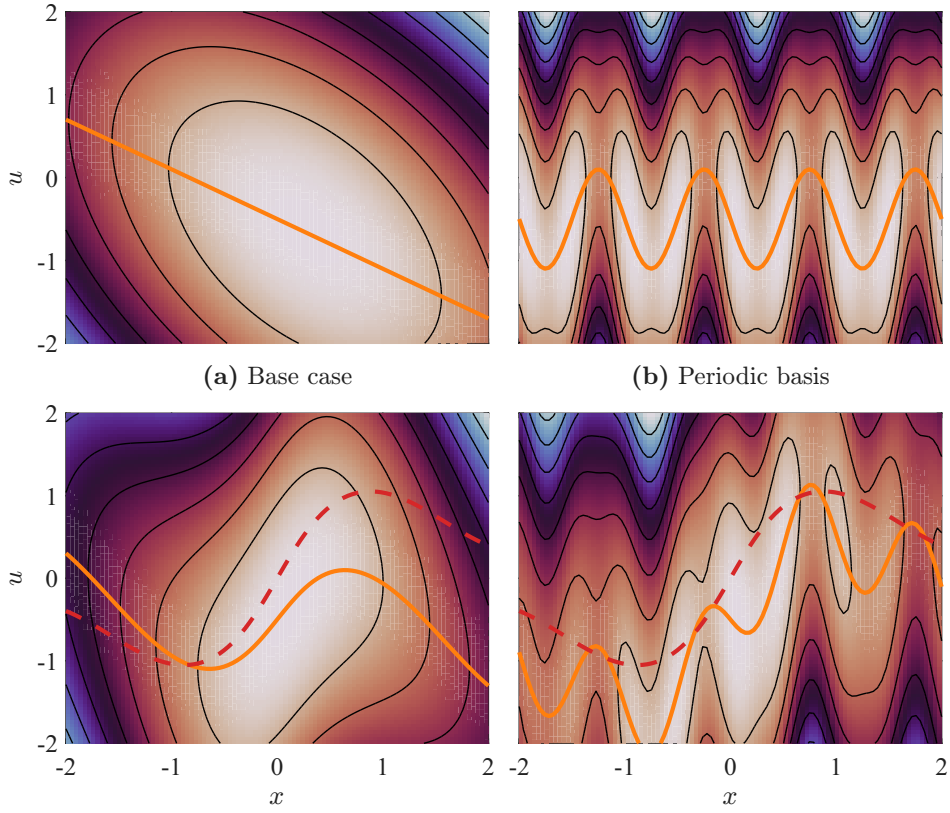
$$\tilde{Q}(x,u,\theta) = V^{\bar{\pi}}(x) + T + \begin{bmatrix}\phi(x)\\ u-\bar{\pi}(x)\end{bmatrix}^\top R - \begin{bmatrix}\phi(x)\\ u-\bar{\pi}(x)\end{bmatrix}^\top S \begin{bmatrix}\phi(x)\\ (u-\bar{\pi}(x))\end{bmatrix}, \tag{8.10}$$

where $\phi:\mathbb{R}^{n_x}\to\mathbb{R}^{n_\phi}$ is a vector of basis functions in the state variable x and in which $T\in\mathbb{R}$, $R := \begin{bmatrix}R_x\\ R_u\end{bmatrix} \in \mathbb{R}^{n_\phi+n_u}$, and $S := \begin{bmatrix}S_{xx} & S_{xu}\\ S_{xu}^\top & S_{uu}\end{bmatrix} \in \mathbb{R}^{(n_\phi+n_u)\times(n_\phi+n_u)}$ with $S = S^\top \succ 0$. The parameter vector θ contains all the elements of T, R, and the upper-triangular elements of S. Hence, $n_\theta = 1 + n_\phi + n_u + \frac{(n_\phi+n_u)(n_\phi+n_u+1)}{2}$. With the stacked vector $\zeta := \begin{bmatrix}\phi(x)^\top & (u-\bar{\pi}(x))^\top\end{bmatrix}^\top$, the parametrization (8.10) reads $\tilde{Q}(x,u,\theta) = V^{\bar{\pi}}(x) + T + \zeta^\top R - \zeta^\top S\zeta$. Figure 8.1 provides intuition behind the proposed parametrization.

Completing the square in (8.10) gives

$$\tilde{Q}(x,u,\theta) = V^{\bar{\pi}}(x) + T + \tfrac{1}{4}R^\top S^{-1}R - \left(\zeta - \tfrac{1}{2}S^{-1}R\right)^\top S\left(\zeta - \tfrac{1}{2}S^{-1}R\right). \tag{8.11}$$

Since $S\succ 0$, the Q-function is strictly concave in ζ, with unconstrained maximizer $\zeta = \frac{1}{2}S^{-1}R$, so the linear term R sets the location of this maximizer. In particular,



(a) Base case **(b)** Periodic basis

(c) Baseline policy **(d)** Combined

Figure 8.1. Quadratic Q-function parametrization (8.10) for a scalar state x and scalar input u, shown over $x, u \in [-2, 2]$. In every panel the color indicates the value of the fitted Q-function $\tilde{Q}(x, u)$ and the black curves are its level sets. Shown are the greedy policy $\tilde{\pi}(x)$ (——) from (8.12), and the baseline policy $\bar{\pi}(x)$ (- - -). The simplest case **(a)** has identity basis $\phi(x) = x$, zero baseline policy $\bar{\pi}(x) = 0$ and zero baseline value $V^{\bar{\pi}}(x) = 0$; a nonzero linear term $R = [0; -1]$ places the maximizer at $\tilde{\pi}(0) = -0.5$, so the policy does not pass through the origin. The same case with a periodic basis $\phi(x) = \sin(2\pi x)$ **(b)** makes the level sets repeat along x. A baseline policy $\bar{\pi}(x) = 0.7\tanh(2x) + 0.4\sin(2x)$ together with a baseline value $V^{\bar{\pi}}(x) = -0.7x^2 + 0.5\cos(2x)$ and identity basis $\phi(x) = x$ **(c)** bends the ridge of maximizers, and the baseline value is added to the Q-function, reshaping it along x. The periodic basis, the baseline policy and the baseline value are combined in **(d)**.

$S_{uu} \succ 0$ makes $\tilde{Q}$ strictly concave in u for every state x, so the greedy action is unique, as we derive next. If $\phi(x) = x$ (and the baseline $\bar{\pi}$ is affine), then ζ is affine in (x, u), and the Q-function is strictly concave in (x, u) as well whenever the baseline value $V^{\bar{\pi}}$ is concave. For a nonlinear basis ϕ, the map $x \mapsto \phi(x)$ is generally not invertible, so concavity in ζ does not imply a single peak in the original variables (x, u). In the LQR setting with linear dynamics and quadratic reward, the choice $\phi(x) = x$ makes the quality function exactly quadratic in (x, u).

Given a Q-function parametrization, the corresponding greedy action can be found by setting the partial derivatives with respect to u equal to zero and solving for u. For the particular parametrization in (8.10), we obtain

$$\begin{aligned} \tilde{\pi}(x) = & \arg\max_u \tilde{Q}(x, u, \theta) \\ & = \bar{\pi}(x) + \frac{1}{2} S_{uu}^{-1} R_u - S_{uu}^{-1} S_{xu}^\top \phi(x). \end{aligned} \tag{8.12}$$

Note that the basis functions ϕ appear linearly in this greedy policy. In this sense, ϕ allows us to choose an affine feedback parametrization. The choice of ϕ also informs the PE requirements on the dataset $\mathcal{D}$, which is explained in the following remark.

Remark 8.1. *Even though the proposed methods are off-policy, the data-generating policy cannot be selected arbitrarily. We need stochasticity in the policy during the creation of dataset $\mathcal{D}$, such that the inputs persistently excite the basis functions ϕ through the dynamics. In this case, the PE condition is satisfied if the parameter optimization problem* (8.8) *with* (8.10) *has a unique solution. The basis ϕ is chosen by hand. Its choice fixes both the achievable policy class, through the greedy action* (8.12)*, and the persistence-of-excitation requirement on $\mathcal{D}$.*

When we substitute (8.12) into the parametrized Q-function (8.10), we obtain

$$\tilde{Q}(x, \tilde{\pi}(x), \theta) = V^{\bar{\pi}}(x) + T + \begin{bmatrix} \phi(x) \\ 1 \end{bmatrix}^\top \left(\Omega + \Psi S_{uu}^{-1} \Psi^\top\right) \begin{bmatrix} \phi(x) \\ 1 \end{bmatrix} \tag{8.13}$$

with

$$\Omega := \begin{bmatrix} -S_{xx} & \frac{1}{2} R_x \\ \frac{1}{2} R_x^\top & 0 \end{bmatrix}, \quad \Psi := \begin{bmatrix} S_{xu} \\ -\frac{1}{2} R_u^\top \end{bmatrix},$$

which presents the Q-value that follows from the so-called greedy action. With the chosen cost function and Q-function structure, we can move to solutions to the parameter optimization problem. The next section details a convex numerical method to this effect.

8.3.2 Iterative Convex Method

Optimization problems with ℓ_1-norm cost can be equivalently characterized with linear cost and inequalities containing the original cost vector. If we perform this transformation and substitute (8.10) and (8.13) into (8.8) with $p = 1$, we obtain

$$\begin{aligned}
\min_{\theta} \quad & \sum_{k=0}^{N} t_k, \\
\text{s.t.} \quad & t_k \geq \tilde{z}_k, && k \in \{0, 1, \ldots, N\}, \\
& t_k \geq -\left(\tilde{z}_k\right), && k \in \{0, 1, \ldots, N\}, \\
& W = \Psi S_{uu}^{-1} \Psi^\top, \\
& \Omega = \begin{bmatrix} -S_{xx} & \frac{1}{2} R_x \\ \frac{1}{2} R_x^\top & 0 \end{bmatrix}, \\
& S \succ 0,
\end{aligned} \tag{8.14}$$

where $t_k \in \mathbb{R}$ for $k \in \{0, 1, \ldots, N\}$ and $\Omega, W \in \mathbb{R}^{(n_\phi+1)\times(n_\phi+1)}$ are auxiliary variables, and in which

$$\begin{aligned}
\tilde{z}_k := \tilde{Q}(x_k, u_k, \theta) - \Bigg(r_k + \gamma \Bigg(& V^{\bar{\pi}}(x_{k+1}) + T \\
& + \begin{bmatrix} \phi(x_{k+1}) \\ 1 \end{bmatrix}^\top (\Omega + W) \begin{bmatrix} \phi(x_{k+1}) \\ 1 \end{bmatrix} \Bigg) \Bigg).
\end{aligned}$$

This optimization problem has a cost function that is linear in the design variables, linear matrix inequality constraints, along with a single matrix equality constraint $W = \Psi S_{uu}^{-1} \Psi^\top$, which is nonlinear in the design variables.

To make the optimization problem convex, we fix the optimization variables that appear nonlinearly in (8.14) by setting W to arbitrary feasible values $\hat{\Psi}$, $\hat{S}_{uu}$. Additionally, we propose fixing the variable Ω to feasible values $\hat{S}_{xx}$ and $\hat{R}_x$, even though Ω appears linearly in the optimization problem. While this step is not required for convexity, empirical results suggest it often leads to improved solutions.

After these adjustments, the optimization problem becomes

$$\begin{aligned} \min_{\theta} \quad & \sum_{k=0}^{N} t_k, \\ \text{s.t.} \quad & t_k \geq \tilde{z}_k, && k \in \{0, 1, \ldots, N\}, \\ & t_k \geq -\left(\tilde{z}_k\right), && k \in \{0, 1, \ldots, N\}, \\ & W = \hat{\Psi} \hat{S}_{uu}^{-1} \hat{\Psi}^\top, \\ & \Omega = \begin{bmatrix} -\hat{S}_{xx} & \frac{1}{2}\hat{R}_x \\ \frac{1}{2}\hat{R}_x^\top & 0 \end{bmatrix}, \\ & S \succ 0. \end{aligned} \tag{8.15}$$

This optimization problem is convex, so any local minimizer is also a global minimizer. In fact, since the cost is linear in the design variables and the constraints are LMIs, the optimization problem (8.15) is a semidefinite programming problem, for which a wide variety of efficient numerical solvers are available. Note that in numerical implementations, e.g., using SDP solvers, strict inequalities are often replaced by non-strict ones. After solving it, we can update $\hat{S}_{xx}$, $\hat{R}_x$, $\hat{\Psi}$ and $\hat{S}_{uu}$ using the solution θ^* and repeat the optimization process. This implementation resembles coordinate descent since we optimize over a subset of design variables. Fixing the next-state value before refitting the Q-function also makes each iteration closely related to fitted Q-iteration [209], for which error-propagation results relate the closed-loop performance of the resulting policy to the per-iteration Bellman error [214]. These results assume the true Bellman error. For stochastic transitions, the single-sample residual used here is a biased estimate of it [215].

At a fixed point of this iteration, the updated values equal those of the solution, that is, $\hat{S}_{xx} = S_{xx}$, $\hat{R}_x = R_x$, $\hat{\Psi} = \Psi$ and $\hat{S}_{uu} = S_{uu}$. The equality constraint $W = \Psi S_{uu}^{-1} \Psi^\top$ of (8.14) then holds, so the iterate is feasible for the original problem. It need not minimize the original problem, since each solve holds the next-state value fixed and ignores how that value varies with θ. This behavior mirrors fixed-target iterations in system identification, such as the Sanathanan–Koerner iteration [216], whose fixed point likewise need not be a stationary point of the original problem. There are no guarantees that a fixed point is reached after finitely many iterations, yet we observe good results in practice.

The next section details an alternative solution to the optimization problem (8.14) using a convex relaxation.

8.3.3 Convex Relaxation

Consider again the optimization problem in (8.8) with $p = 1$ and $\tilde{Q}$ parametrized according to (8.10). A *convex relaxation* of this problem is given by the optimization

problem

$$\begin{aligned}
\min_{\theta} \quad & \sum_{k=0}^{N} t_k \\
\text{s.t.} \quad & t_k \geq \tilde{z}_k, && k \in \{0, 1, \ldots, N\}, \\
& t_k \geq -\left(\tilde{z}_k\right), && k \in \{0, 1, \ldots, N\}, \\
& \begin{bmatrix} W & \Psi \\ \Psi^\top & S_{uu} \end{bmatrix} \succeq 0, \\
& \Omega = \begin{bmatrix} -S_{xx} & \frac{1}{2} R_x \\ \frac{1}{2} R_x^\top & 0 \end{bmatrix}, \\
& S \succ 0.
\end{aligned} \tag{8.16}$$

We obtain this relaxation by replacing the equality constraint $W = \Psi S_{uu}^{-1} \Psi^\top$ by the inequality constraint $W \succeq \Psi S_{uu}^{-1} \Psi^\top$. Afterward, since $S \succ 0$ implies that $S_{uu} \succ 0$, we can apply the Schur complement. We obtain the equivalent matrix inequality included as a constraint in (8.16) which is linear in the design parameters. The resulting *relaxed* optimization problem (8.16) is a (convex) semidefinite programming (SDP) problem, as the cost is linear in the design variables and the constraints are LMIs. Furthermore, it is a relaxation of (8.14) since the cost function is unchanged and the feasible domain is (slightly) expanded.

The Schur complement step above uses only $S_{uu} \succ 0$, so the full constraint $S \succ 0$ is not required for convexity of the reformulation, nor for the uniqueness of the greedy action (8.12). We retain $S \succ 0$ as a modeling choice, since it makes the parametrized Q-function strictly concave in the stacked vector ζ.

The convex relaxation provides guaranteed lower and upper bounds on the optimal cost. The lower bound arises because the relaxed optimal cost can never exceed the original, as the relaxation expands the feasible set. Moreover, the relaxed solution for θ is feasible for the original problem (8.14) if we set $W = \Psi S_{uu}^{-1} \Psi^\top$, allowing us to evaluate an upper bound. If the bounds are tight, then the relaxed solution is also optimal for the original problem.

In order to make the relaxation less conservative, we can add the matrix inequality

$$S_{xx} \succ W_{11}, \tag{8.17}$$

where $W_{11} \in \mathbb{R}^{n_\phi \times n_\phi}$ is the top left submatrix of W. This matrix inequality follows from the original non-relaxed optimization problem by taking the Schur complement of S. The introduction of the matrix inequality (8.17) constrains W from above, potentially decreasing the gap between W and $\Psi S_{uu}^{-1} \Psi^\top$.

Next, we introduce a penalty term to the cost function, such that the inequality $W \succeq \Psi S_{uu}^{-1} \Psi^\top$ becomes closer to equality. This penalty term can be chosen in

many different ways. In this chapter, we choose the nuclear norm of W, i.e.,

$$\|W\|_* := \sum_{i=1}^{n_\phi+1} \sigma_i \tag{8.18}$$

where σ_i are the singular values of W. Note that since W is square and positive semidefinite, $\|W\|_* = \operatorname{tr}(W)$.

The choice of the nuclear norm as the penalty function is motivated by several factors. First, the trace operator maintains the convexity of the optimization problem. Second, penalizing the nuclear norm — equal to the sum of the eigenvalues of W — directly addresses the gap between W and $\Psi S_{uu}^{-1}\Psi^\top$ by discouraging excessively large eigenvalues of W. Last, in typical applications we see that $n_u < n_\phi + 1$, which results in $\Psi S_{uu}^{-1}\Psi^\top$ being rank-deficient. The nuclear norm presents a convexification of matrix rank minimization [217] and can therefore be used to promote rank-deficient W. This combination of a Schur-complement relaxation with a trace penalty resembles the cone-complementarity linearization of El Ghaoui et al. [218]. Conditions under which such nuclear-norm penalization recovers the exact low-rank solution, and hence closes the relaxation gap, are known in special cases [217].

Together with the previous idea, this leads to the following optimization problem

$$\begin{aligned}
\min_{\theta} \quad & \sum_{k=0}^{N} t_k + \lambda \operatorname{tr}(W), \\
\text{s.t.} \quad & t_k \geq \tilde{z}_k, && k \in \{0, 1, \dots, N\}, \\
& t_k \geq -(\tilde{z}_k), && k \in \{0, 1, \dots, N\}, \\
& \begin{bmatrix} W & \Psi \\ \Psi^\top & S_{uu} \end{bmatrix} \succeq 0, \\
& \Omega = \begin{bmatrix} -S_{xx} & \frac{1}{2}R_x \\ \frac{1}{2}R_x^\top & 0 \end{bmatrix}, \\
& S \succ 0, \\
& S_{xx} \succ W_{11},
\end{aligned} \tag{8.19}$$

Given a dataset $\mathcal{D}$, the penalty weight λ can be chosen by line search to minimize the original cost (8.14). Section 8.4.3 demonstrates this for the pendulum.

8.3.4 Algorithms

Combining all the ideas presented, we can summarize the LMI-based parametrized Q-learning methods in two pseudocode algorithms. **Algorithm 2** details the LMI-based Q-learning method that uses convex relaxation (LMI-QL), while **Algorithm 3**

details the iterative LMI-based Q-learning method that resembles coordinate descent (LMI-QLi), with $\tau \in \mathbb{N}$ the number of iterations.

Algorithm 2 LMI-QL

Choose $\bar{\pi}$ along with an (approximate) value function $V^{\bar{\pi}}$
Generate $\mathcal{D}$ using an arbitrary (stochastic) policy
repeat
    Choose λ using line search
    Find θ^*, the minimizing solution to (8.19)
until the line search stopping condition is reached

Algorithm 3 LMI-QLi

Choose $\bar{\pi}$ along with an (approximate) value function $V^{\bar{\pi}}$
Choose $\tau \in \mathbb{N}$
Generate $\mathcal{D}$ using an arbitrary (stochastic) policy
Initialize $\hat{S}_{xx}$, $\hat{R}_x$, $\hat{\Psi}$, $\hat{S}_{uu}$
repeat
    Find θ^*, the minimizing solution to (8.15)
    Update $\hat{S}_{xx}$, $\hat{R}_x$, $\hat{\Psi}$, $\hat{S}_{uu}$ using θ^*
    $\tau \leftarrow \tau - 1$
until $\tau = 0$

The two methods can be used to complement each other. We can use LMI-QL to obtain a lower bound on the optimal ℓ_1 cost, and then use the solution to warm-start LMI-QLi. In terms of computational cost, both methods reduce to semidefinite programs whose size grows linearly in the dataset size, with $O(N)$ variables and constraints. LMI-QL solves one such program, whereas LMI-QLi solves τ of them, and is therefore a factor τ more expensive.

In the next section, we will demonstrate the merit of the proposed method in a simulation case study.

8.4 Pendulum System Simulation Case Study

We apply the proposed methods to a nonlinear pendulum and learn the deviation from a given baseline policy. The goal is to stabilize the pendulum in the upright position.

8.4.1 Simulated Setup

The pendulum has mass m and length l and is affected by gravity with gravitation constant g. Furthermore, the pendulum is subject to linear damping proportional to the angular velocity with damping constant d. The pendulum equations are discretized in time using the forward Euler method with a sample time of $T_s = 0.01$. The state representation of the pendulum consists of its angle and angular velocity. The dynamical model after discretization is given by

$$x_{k+1} = \begin{bmatrix} \operatorname{mod}\left(x_k[1] + T_s x_k[2] + \pi, 2\pi\right) - \pi \\ x_k[2] + \frac{T_s g}{l} \sin\left(x_k[1]\right) - \frac{T_s d}{ml^2} x_k[2] + \frac{T_s}{ml^2} u_k \end{bmatrix} + w_k, \tag{8.20}$$

where $m = 0.1, l = 1, g = 9.81, d = 0.1$, and with w_k as an independent and identically distributed (i.i.d.) zero-mean Gaussian disturbance with covariance matrix $\Sigma_w = 2.5 \cdot 10^{-5} I_{n_x}$. The reward function is quadratic, i.e.,

$$r_k = -\begin{bmatrix} x_k^\top & u_k^\top \end{bmatrix} M \begin{bmatrix} x_k \\ u_k \end{bmatrix}, \tag{8.21}$$

in which $M = M^\top \succ 0$. The discount factor in (8.3) is given by $\gamma = 0.98$. We use an exploring policy, namely, $u_k \sim \mathcal{N}(0, 10^1 I_{n_u})$ at every timestep k. The state values in the dataset $\mathcal{D}$ are subjected to i.i.d. zero-mean Gaussian noise with covariance matrix $\Sigma_v = 10^{-4} I_{n_x}$. We choose

$$\phi(x) = \begin{bmatrix} x \\ \sin(x[1]) \end{bmatrix},$$

which allows our controller to counteract the nonlinear effects of gravity on the pendulum. The baseline controller $\bar{\pi}$ is an LQR state feedback controller based on a linearization of (8.20) around the origin, with inaccurate parameters $\bar{m} = 0.1m$ and $\bar{l} = 0.3l$. We set $R = 0$, which by (8.11) places the maximizer of the Q-function at $\zeta = 0$, that is, at the origin $x = 0$ with input $u = \bar{\pi}(0) = 0$. This is where the reward (8.21) is maximized in this specific case study.

The baseline value function is selected as $V^{\bar{\pi}}(x) = -x^\top \bar{P} x$, with $\bar{P}$ the solution to the algebraic Riccati equation for the inaccurate linearized model. We compare the proposed method to a model-based controller consisting of feedback linearization and LQR on the resulting linear model (with accurate model parameters). This controller is a strong model-based baseline, as it uses the accurate model parameters and enjoys an inverse-optimality property [219]. We also compare the proposed method to the learning method least-squares policy iteration (LSPI), which is detailed in the next section.

8.4.2 Least-Squares Policy Iteration

In LSPI, we perform policy evaluation steps called LSTD-Q, followed by policy improvement steps based on the updated Q-function. The Q-function parametrization

in LSPI is linear in the parameters, i.e.,

$$\tilde{Q}^{LSPI}(x, u, \theta) = \psi^\top(x, u)\theta^{LSPI}, \tag{8.22}$$

with $\theta^{LSPI} \in \mathbb{R}^{n_\theta}$, and $\psi(x, u) := \begin{bmatrix}\psi_1(x, u), & \psi_2(x, u), & \ldots, & \psi_{n_\theta}(x, u)\end{bmatrix}^\top$ a vector of basis functions, with $\psi_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$, $i \in \{1, 2, \ldots, n_\theta\}$. Note that this Q-function parametrization slightly generalizes (8.10). The linearity of $\tilde{Q}^{LSPI}$ in θ^{LSPI} allows us to reformulate the policy evaluation step as a convex least-squares problem with a closed-form solution by optimizing the parameters using (8.8) with $p = 2$. Importantly, this requires that we fix the policy at the next timestep. In the simulated case study, we use the same Q-function parametrization for LSPI as in the proposed method, i.e., (8.10). Note that LSPI cannot include a positive definite constraint on S.

8.4.3 Tuning the Penalty λ

We choose the penalty weight λ in (8.19) by line search. For each λ, we solve the relaxed problem and evaluate the original cost (8.14) at the relaxed solution, and we keep the λ that minimizes this original cost. Figure 8.2 shows the lower and upper bounds on the optimal ℓ_1 cost as functions of λ for the pendulum dataset, together with the closed-loop reward. The line search is performed offline. Automating the choice of λ is a direction for future work.

8.4.4 Results

Figure 8.3 shows the results of applying LSPI, feedback linearization + LQR, and LMI-QL(i) to the pendulum example. We construct a dataset $\mathcal{D}$ with 500 randomly generated initial states, stochastic inputs, rewards, and next states. The learning methods then learn from zero using progressively larger subsets of $\mathcal{D}$ to assess learning speed. For LSPI and LMI-QLi, we run 20 iterations for each subset. The methods are compared based on cumulative reward over a 100-sample simulation with identical initial conditions and disturbances. To account for variability, we perform 500 Monte Carlo runs. Runs whose cumulative reward falls below -2000 are treated as unstable and excluded before computing the median and percentile band, which affects 20.6% of the LSPI runs but only 0.83% of the LMI-QL runs and 0.27% of the LMI-QLi runs.

We also assess the relaxation gap on the full dataset of $N = 500$ samples. Taking the median over the 500 seeds at a fixed penalty weight $\lambda = 0.02$, LMI-QL attains a relaxed cost of 3.602, a lower bound on the optimal ℓ_1 cost, and an original cost of 4.042 at the relaxed solution, an upper bound. The median of the per-run relative gap (upper $-$ lower)/upper is 10.3%. The gap is dataset-dependent: about a third of the Monte Carlo runs are nearly tight, with a gap below 1%, while around a

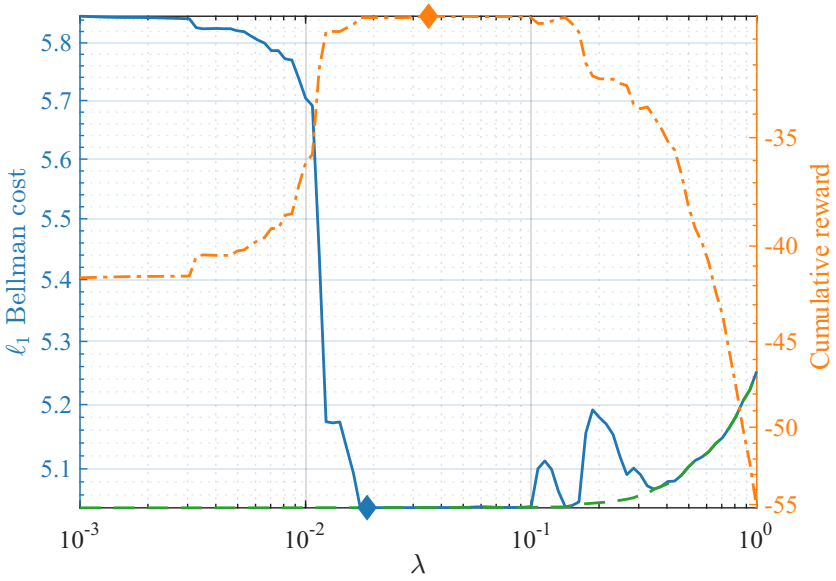


Figure 8.2. Effect of the penalty weight λ on the pendulum dataset. The left axis shows the upper bound (—, original cost at the relaxed solution) and lower bound (- - -, relaxed cost) on the optimal ℓ_1 Bellman cost, and the right axis shows the closed-loop cumulative reward (-·-·-). All axes use a logarithmic scale. The blue marker (◆) indicates the λ that minimizes the upper bound and the orange marker (◆) the λ that maximizes the reward, which align closely.

fifth exceed 20%. The near-coincident bounds in Figure 8.2 reflect a dataset on which λ was directly tuned, which results in a smaller gap. LMI-QLi attains a median original cost of 3.777, which presents a tighter upper bound than LMI-QL.

In Figure 8.3, we can observe that both proposed methods converge close to the cumulative reward that feedback linearization + LQR can achieve. At zero data used, LMI-QL and LMI-QLi start from the baseline policy $\bar{\pi}$, while LSPI starts from the zero-input policy. This baseline head start lets the proposed methods begin ahead of LSPI before any data has been used, since the baseline provides a good starting point for the Q-function values. Lastly, observe that the proposed methods converge faster than LSPI. This can be explained by the fact that the LSPI solution does not satisfy the $S \succ 0$ constraint at multiple points in Figure 8.3, and by the fact that the proposed methods more directly learn the optimal parametrized Q-function.

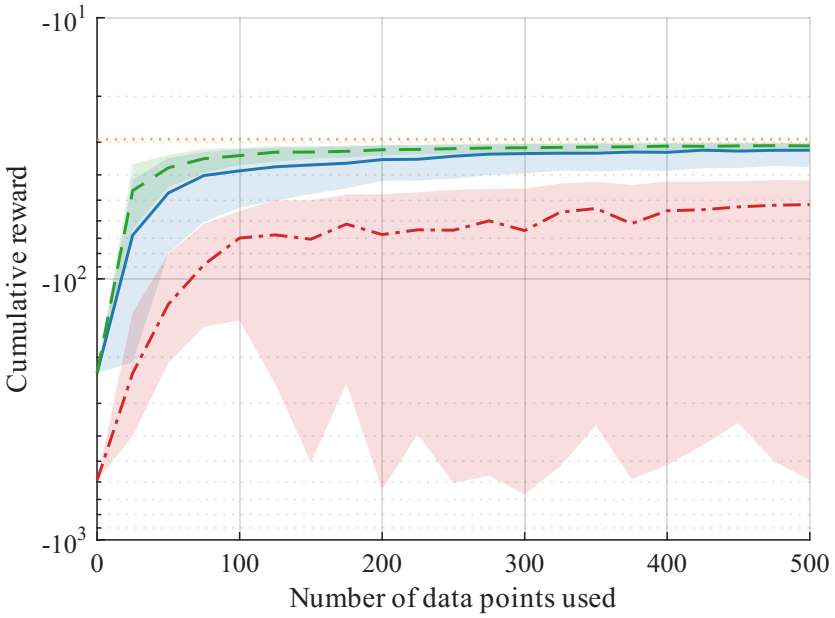


Figure 8.3. Median cumulative reward and 25th–75th percentile band over 500 Monte Carlo runs of a 100-sample simulation, against the number of data points used, with the reward on a logarithmic axis. Runs with cumulative reward below −2000 are excluded as unstable. We compare LQR with feedback linearization (·····), LMI-QL (——), LMI-QLi (– – –) and LSPI (–·–·–).

8.5 Conclusions

This chapter introduces LMI-QL and LMI-QLi, two methods for direct optimal Q-function learning using convex optimization. We show that if the Q-function is parametrized in a quadratic form and if the objective function uses the ℓ_1-norm of the Bellman residuals, we can derive two separate convex optimization problems. In LMI-QLi, we fix a subset of the design variables such that the optimization problem for the remaining variables becomes a semidefinite program. In LMI-QL, we relax the temporal difference minimization problem into a semi-definite program including a set of linear matrix inequalities. Simulated results indicate high data efficiency in comparison to least-squares policy iteration for a nonlinear example.

Several directions for future work can be pursued. Establishing convergence properties of LMI-QLi, and conditions under which its fixed point is a stationary point of (8.14), are of interest. For LMI-QL, sufficient conditions under which the convex relaxation is exact, and how they relate to the value of λ at which the

gap closes, are a natural next step. The connection between the Bellman residual and the value-function error from Section 8.3 is idealized. A finite-sample version that holds for the estimated residual, under a persistence-of-excitation assumption, would bound the distance to the optimal policy. A further direction is output feedback, where the full state is not measured.

9

Smart Exploration in Reinforcement Learning Using Bounded Uncertainty Models

Abstract - Reinforcement learning (RL) is a powerful framework for decision-making in uncertain environments, but it often requires large amounts of data to learn an optimal policy. We address this challenge by incorporating prior model knowledge to guide exploration and accelerate the learning process. Specifically, we assume access to a model set that contains the true transition kernel and reward function. We optimize over this model set to obtain upper and lower bounds on the Q-function, which are then used to guide the exploration of the agent. We provide theoretical guarantees on the convergence of the Q-function to the optimal Q-function under the proposed class of exploring policies. Furthermore, we also introduce a data-driven regularized version of the model set optimization problem that ensures the convergence of the class of exploring policies to the optimal policy. Lastly, we show that when the model set has a specific structure, namely the bounded-parameter MDP (BMDP) framework, the regularized model set optimization problem becomes convex and simple to implement. In this setting, we also prove finite-time convergence to the optimal policy under mild assumptions. We demonstrate the effectiveness of the proposed exploration strategy, which we call **BUMEX** (*Bounded Uncertainty Model-based Exploration*), in a simulation study. The results indicate that the proposed method can significantly accelerate learning

This chapter is based on van Hulst et al. [220].

in benchmark examples. A toolbox is available at `https://github.com/JvHulst/BUMEX`.

9.1 Introduction

Reinforcement learning (RL) has shown great success in solving complex decision-making problems in various domains, such as robotics, games, and finance [202]. However, RL often requires large amounts of data to learn an optimal policy, which can be impractical in many real-world applications. This is especially true in environments where data collection is expensive or time-consuming, such as in robotics or healthcare. The need for large datasets comes from the fact that RL algorithms typically explore the environment randomly, causing the agent to visit many suboptimal states and actions [221].

A key opportunity arises when one has access to model knowledge, which is often the case in many scenarios. This information is typically not utilized in traditional RL algorithms. For instance, we may know that certain states cannot be reached from particular other states or that there is a specific goal state. In this chapter, we propose using such available knowledge, giving information on the set of models to which the true model belongs, and using this to guide the agent's exploration, thus speeding up the learning process.

More specifically, we propose estimating bounds on the Q-function using optimistic/pessimistic optimization over the available model set. We then use these bounds to guide the agent's exploration by prioritizing uncertain or promising regions of the state/action space. Additionally, we propose regularizing the model set optimization problem using a database of observed transitions. We show that the Q-function converges to the optimal Q-function under the exploring policy and that the exploring policy converges to the optimal policy in the regularized case. Moreover, for a specific model set characterization we obtain a practical algorithm with a convex (regularized) model set optimization. We refer to this practical algorithm as BUMEX (*Bounded Uncertainty Model-based Exploration*).

This work is at the intersection of model-based RL and smart exploration in RL. We will now briefly review the relevant literature, highlighting the relation and differences to the current work.

Model-based reinforcement learning is a popular approach to speed up the learning process in RL by incorporating model knowledge. In traditional model-based RL, the agent learns a model of the environment and uses this model to plan its actions [222, 223]. In our work, we use prior model knowledge to guide the exploration of the agent, rather than to plan its actions. Smart exploration in RL focuses on how to explore the environment efficiently to speed up learning. Several prominent approaches use confidence bounds on value functions to guide exploration, including Upper Confidence Bound methods [221, 224], UCRL2 [225], and Posterior Sampling for Reinforcement Learning (PSRL) [226]. Bayesian RL builds on this by using Bayesian inference to guide the agent's exploration [227]. These methods typically rely on uncertainty estimates derived from a single nominal

model or confidence intervals around model parameters. In contrast, our work employs a set of models to direct exploration, rather than relying on a nominal model with uncertainty estimates.

Our work builds most directly on the bounded-parameter MDP (BMDP) framework [228], which characterizes model uncertainty through intervals on transition probabilities and rewards. BMDPs are closely related to uncertain MDPs (uMDPs) and interval MDPs (IMDPs). The BMDP framework has been used to develop algorithms that are robust to model uncertainty [229]. The present work builds upon this idea by using the BMDP framework to guide the exploration of the agent. Additionally, our exploration strategy draws inspiration from Bayesian optimization [72], which balances exploration and exploitation by using acquisition functions. We adapt similar heuristics for selecting actions based on Q-function bounds.

The main contributions can be summarized as follows.

- We introduce an exploration strategy for reinforcement learning that leverages bounds on the Q-function obtained from a model set. We provide theoretical guarantees on the convergence of the Q-function to the optimal Q-function under the proposed exploration strategy.
- We propose a data-regularized version of the model set optimization problem that ensures convergence of the Q-function bounds to the optimal Q-function.
- We propose a practical algorithm, BUMEX, for the BMDP framework and establish, under specific conditions, finite-time convergence to the optimal policy.

Our main theoretical results focus on finite state-action spaces, which allows us to provide (finite-time) convergence guarantees and construct a practical algorithm.

This chapter is organized as follows. Section 9.2 introduces the problem formulation. Section 9.3 presents the general framework for the proposed method and the technical results. Section 9.4 discusses a practical algorithm that leverages the theoretical results. Section 9.5 presents simulations and Section 9.6 concludes the chapter.

9.2 Problem Formulation

We model the decision-making environment as a Markov decision process (MDP), which we define in a general setting to encompass both finite and infinite (continuous) state and action spaces. We assume an infinite-horizon discounted MDP, where the agent interacts with the environment over an infinite number of time steps. This MDP is defined as the tuple $\mathcal{M} = (\mathcal{X}, \mathcal{U}, P, g, \gamma)$, with the following ingredients:

- **State Space** $\mathcal{X}$ is a measurable (Polish) space representing the set of possible states of the environment. We denote the Borel σ-algebra on $\mathcal{X}$ by $\mathcal{B}(\mathcal{X})$.
- **Action Space** $\mathcal{U}$ is a measurable (Polish) space representing the set of inputs or actions that the agent can take. We denote the Borel σ-algebra on $\mathcal{U}$ by $\mathcal{B}(\mathcal{U})$.
- **Transition Kernel** $P : \mathcal{X} \times \mathcal{U} \times \mathcal{B}(\mathcal{X}) \to [0, 1]$ is a Markovian transition kernel, where $P(x, u, \cdot)$ is a probability measure on $\mathcal{X}$ for all $(x, u) \in \mathcal{X} \times \mathcal{U}$. We assume that for every bounded measurable function $h : \mathcal{X} \to \mathbb{R}$, the mapping $(x, u) \mapsto \int_{\mathcal{X}} h(x')P(x, u, dx')$ is measurable.
- **Reward Function** $g : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is a bounded, measurable function, where $g(x, u)$ is the reward obtained by the agent when taking action u in state x. We assume deterministic rewards for simplicity. The extension to stochastic rewards is straightforward.
- **Discount Factor** $\gamma \in [0, 1)$ is the discount factor that determines the importance of future rewards.

The goal in this setting is to find a policy $\pi \in \Pi$, where $\Pi := \{\pi : \mathcal{X} \to \text{Prob}(\mathcal{U}) \mid \pi \text{ is measurable}\}$, such that the expected cumulative reward or value is maximized. Here $\text{Prob}(\mathcal{U})$ denotes the set of Borel probability measures on $(\mathcal{U}, \mathcal{B}(\mathcal{U}))$. The value function $V^\pi : \mathcal{X} \to \mathbb{R}$ of a policy $\pi \in \Pi$ is defined as

$$V^\pi(x) = \mathbb{E}^\pi \left[\sum_{k=0}^{\infty} \gamma^k g(x_k, u_k) \,\middle|\, x_0 = x \right], \tag{9.1}$$

where (x_k, u_k) denotes the state-action pair at time k under policy π, and $\mathbb{E}^\pi$ denotes the expectation with respect to the probability measure induced by π and the transition kernel P. A policy's Q-function $Q^\pi : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is defined as

$$Q^\pi(x, u) = g(x, u) + \gamma \int_{\mathcal{X}} V^\pi(x')P(x, u, dx'), \tag{9.2}$$

where $Q^\pi(x, u)$ is the expected cumulative reward of taking action u in state x, then following policy π. We assume that for every $\pi \in \Pi$, the integrals in (9.1), (9.2) are well-defined. The optimal Q-function $Q^* : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is defined as

$$Q^*(x, u) = \sup_{\pi \in \Pi} Q^\pi(x, u). \tag{9.3}$$

An optimal policy π^* can be obtained from the optimal Q-function as $\pi^*(x) \in \arg\max_{u \in \mathcal{U}} Q^*(x, u)$. The optimal Q-function can also be defined in terms of the Bellman operator $\mathcal{T}$ as $Q^* = \mathcal{T}Q^*$, where $\mathcal{T} : \mathcal{B}(\mathcal{X} \times \mathcal{U}) \to \mathcal{B}(\mathcal{X} \times \mathcal{U})$ is defined as

$$(\mathcal{T}Q)(x, u) = g(x, u) + \gamma \int_{\mathcal{X}} \sup_{u' \in \mathcal{U}} Q(x', u')P(x, u, dx').$$

As usual in reinforcement learning, our objective is to find the optimal policy π^* by interacting with the environment and learning from the observed data $\mathcal{D} := \{(x_k, u_k, x_{k+1})\}_{k=0}^{N-1}$. The transition kernel P and the reward function g are assumed to be unknown to the agent. However, we assume that the agent has access to a model set $\hat{\mathcal{M}}$ that contains the true transition kernel and reward function. This is formalized in the following assumption.

Assumption 9.1. The model set $\hat{\mathcal{M}}$ is defined as $\hat{\mathcal{M}} = (\mathcal{X}, \mathcal{U}, \mathcal{P}, \mathcal{G}, \gamma)$, where $\mathcal{P}$ is a set of transition kernels and $\mathcal{G}$ is a set of reward functions. The true transition kernel P belongs to $\mathcal{P}$, and the true reward function g belongs to $\mathcal{G}$. The model set $\hat{\mathcal{M}}$ is known to the agent.

Thus, the problem is to find the optimal policy π^* based on the model set $\hat{\mathcal{M}}$ and the observed data $\mathcal{D}$.

9.3 Proposed Method

We will first discuss a version of the proposed method that fits the general MDP setting introduced in the previous section. This version cannot be directly applied in practice because optimization over the general model set is intractable. This version presents the theoretical foundation for a more practical version that we will detail in Section 9.4.

9.3.1 General Framework

Given the model set, we can determine bounds on the Q-function that hold for all models in the set. These bounds are constructed using the Bellman updates, given by

$$\begin{aligned}(\underline{\mathcal{T}}\underline{Q})(x,u) &= \inf_{g\in\mathcal{G}} g(x,u) + \gamma \inf_{P\in\mathcal{P}} \int_{\mathcal{X}} \underline{V}(x')P(x,u,dx'),\\ (\bar{\mathcal{T}}\bar{Q})(x,u) &= \sup_{g\in\mathcal{G}} g(x,u) + \gamma \sup_{P\in\mathcal{P}} \int_{\mathcal{X}} \bar{V}(x')P(x,u,dx'),\end{aligned} \tag{9.4}$$

in which $\underline{V}$ and $\bar{V}$ are defined as

$$\underline{V}(x) := \sup_{u\in\mathcal{U}} \underline{Q}(x,u), \quad \bar{V}(x) := \sup_{u\in\mathcal{U}} \bar{Q}(x,u). \tag{9.5}$$

These relationships and the resulting bounds are related to adversarial MDPs, where the agent aims to minimize the reward in the worst-case scenario, see e.g. [230]. We assume throughout that the suprema and infima in (9.4) are attained and that the resulting functions are measurable, so that $\underline{\mathcal{T}}$ and $\bar{\mathcal{T}}$ map $\mathcal{B}(\mathcal{X}\times\mathcal{U})$ into itself.

Both conditions hold automatically when $\mathcal{X}$ and $\mathcal{U}$ are finite, which is the setting of every convergence result in this chapter. In the general case, sufficient conditions follow from the measurable selection theory of Bertsekas and Shreve [231, Ch. 7].

Lemma 9.2. *Let $\underline{\mathcal{T}}$ and $\bar{\mathcal{T}}$ denote the Bellman operators defined in* (9.4)*. These operators are contraction mappings on $\mathcal{B}(\mathcal{X} \times \mathcal{U})$ with unique fixed points $\underline{Q}^*$ and $\bar{Q}^*$, respectively. Then*

$$\underline{Q}^*(x,u) \leq Q^*(x,u) \leq \bar{Q}^*(x,u), \quad \forall (x,u) \in \mathcal{X} \times \mathcal{U}, \tag{9.6}$$

where Q^* is the optimal Q-function defined in (9.3)*. Similarly, the corresponding value functions*

$$\underline{V}^*(x) := \sup_{u \in \mathcal{U}} \underline{Q}^*(x,u) \quad \textit{and} \quad \bar{V}^*(x) := \sup_{u \in \mathcal{U}} \bar{Q}^*(x,u),$$

satisfy $\underline{V}^*(x) \leq V^*(x) \leq \bar{V}^*(x), \quad \forall x \in \mathcal{X}$, where $V^*(x) := \sup_{u \in \mathcal{U}} Q^*(x,u)$ is the optimal value function.

Proof. Since the reward function g is bounded and $\gamma \in [0,1)$, the operators $\mathcal{T}$ and $\bar{\mathcal{T}}$ are contraction mappings with modulus γ with respect to the supremum norm. By the Banach fixed-point theorem, each operator has a unique fixed point. Since the true MDP $\mathcal{M}$ is contained in the model set $\hat{\mathcal{M}}$, we have $\underline{\mathcal{T}} Q \leq \mathcal{T} Q \leq \bar{\mathcal{T}} Q$ for any Q-function Q. The monotonicity of these operators implies the ordering of their fixed points as in (9.6), and the value function bounds follow directly. □

The paper Givan et al. [228] on bounded-parameter MDPs establishes similar bounds for the value function in finite state spaces using interval-based model sets. Our result generalizes this by considering both finite and infinite spaces and by using Q-function bounds instead. Note that the initialization of the Q-bounds does not affect the sandwiching property that is obtained upon convergence. Typically, we initialize the Q-bounds using prior knowledge or heuristics about the environment to speed up the convergence.

As stated next, the guaranteed Q-bounds can be used to test if certain inputs are suboptimal or optimal.

Proposition 9.3. *Let $\underline{Q}^*$ and $\bar{Q}^*$ be guaranteed lower and upper bounds on the optimal Q-function Q^*, respectively, i.e.,* (9.6) *holds. Then, the input u at state x is suboptimal if*

$$\bar{Q}^*(x,u) < \underline{V}^*(x), \tag{9.7}$$

and the input u at state x is optimal if

$$\underline{Q}^*(x,u) \geq \max_{v \in \mathcal{U} \setminus u} \bar{Q}^*(x,v). \tag{9.8}$$

This result follows directly from the guaranteed Q-bounds.

In the next section, we will discuss an exploration strategy that leverages the Q-function bounds to guide the agent's exploration in the environment.

9.3.2 Exploration Strategy

We propose a class of exploring policies to be used during the acquisition of the data that leverages the Q-function bounds. The exploring policy randomly selects the input u at state x according to weighted random sampling with input-dependent weights

$$\phi(x,u) = \begin{cases} \xi, & \text{if } \underline{Q}(x,u) \geq \max_{v \in \mathcal{U} \setminus \{u\}} \bar{Q}(x,v), \\ \beta(x,u), & \text{if } \underline{Q}(x,u) \neq \bar{Q}(x,u) \text{ and } \bar{Q}(x,u) > \underline{V}(x), \\ 0, & \text{if } \underline{Q}(x,u) = \bar{Q}(x,u) \text{ and } \bar{Q}(x,u) \leq \underline{V}(x), \\ \zeta, & \text{if } \underline{Q}(x,u) \neq \bar{Q}(x,u) \text{ and } \bar{Q}(x,u) \leq \underline{V}(x), \\ \zeta, & \text{if } \underline{Q}(x,u) = \bar{Q}(x,u) \text{ and } \bar{Q}(x,u) > \underline{V}(x), \end{cases} \tag{9.9}$$

with $\phi : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_{\geq 0}$ the sampling weight of input u at state x. Here, $\xi \in \mathbb{R}_{>0}$ and $\zeta \in \mathbb{R}_{\geq 0}$ are constants, and $\beta : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_{>0}$. The conditions are evaluated in the order listed, and the first one that applies determines the weight. The induced exploration policy π^ϕ is defined by

$$\pi^\phi(x,u) = \frac{\phi(x,u)}{\int_{\mathcal{U}} \phi(x,v)\, \mu(dv)}, \tag{9.10}$$

with μ a suitable reference measure on $\mathcal{U}$ (e.g., the counting measure if $\mathcal{U}$ is finite).

The first case in (9.9) implies that the input u at state x is guaranteed to be optimal (by Proposition 9.3). For this reason, we assign it an arbitrary positive weight ξ. Note that if an input satisfies this condition, this rules out the possibility that any other inputs satisfy the second condition in (9.9).

The second case implies that the quality of input u at state x is uncertain but that the input has the potential to improve $\underline{V}$. We assign this input a positive weight β that depends on x and u. This allows us to potentially assign more weight to the input if it is highly uncertain or has a high Q-value upper bound $\bar{Q}$. Section 9.3.2 provides heuristic guidance on the choice of β, inspired by Bayesian optimization.

The third case in (9.9) assigns zero weight to actions with tight Q-bounds that are guaranteed to be suboptimal. The fourth case assigns a nonnegative weight of ζ to inputs which are uncertain yet appear to be suboptimal, while the fifth case assigns the same weight to inputs which are certain and not guaranteed to be suboptimal. Note that in many cases, we can select $\zeta = 0$ to prune such inputs. However, this pruning might be too aggressive in some cases if we introduce data-driven regularization of the model set optimization, which will be discussed later in this chapter. Figure 9.1 illustrates the exploring policy based on the bounds.

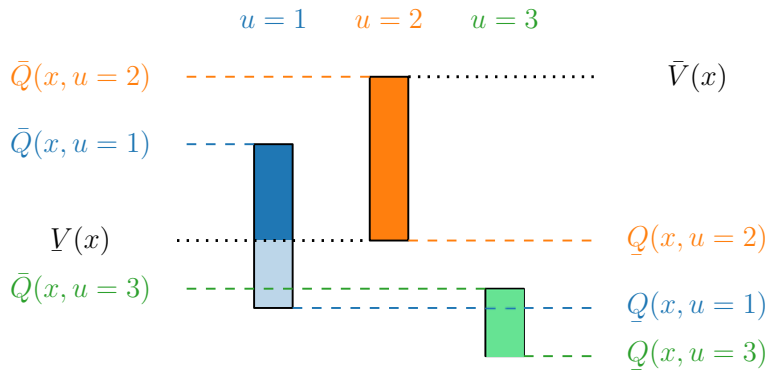


Figure 9.1. Schematic representation of Q-function bounds. From this particular example, using Proposition 9.3 we can deduce that the choice $u = 3$ is guaranteed to be suboptimal for this state x, as its upper bound Q-value $\bar{Q}(x, 3)$ is lower than $\underline{V}(x)$. By (9.9), the input choice $u = 3$ is therefore assigned a weight of ζ. In contrast, actions $u = 1$ and $u = 2$, for which the bounds do not decisively rule out optimality, receive weight $\beta(x, u)$. Since $\underline{Q}(x, 1)$ is below $\underline{V}(x)$, the proposed heuristics in Section 9.3.2 explore the input $u = 2$ more frequently.

As shown next, the convergence properties of standard Q-learning are retained by the proposed exploring policy, as the exploration is based on guaranteed bounds.

Theorem 9.4. *Consider finite state-action spaces. Let Q be the Q-function obtained using standard Q-learning*

$$Q(x_k, u_k) \leftarrow (1 - \alpha_k) Q(x_k, u_k) + \alpha_k \left[r_k + \gamma \max_{u' \in \mathcal{U}} Q(x_{k+1}, u') \right], \tag{9.11}$$

under the exploring policy π^ϕ defined in (9.10) with parameters $\xi \in \mathbb{R}_{>0}, \zeta \in \mathbb{R}_{\geq 0}$ and function $\beta : \mathcal{X} \times \mathcal{U} \to \mathbb{R}_{>0}$, initialized such that $\underline{Q} \leq Q \leq \bar{Q}$. If every state is visited infinitely often (through e.g., exploring starts), and the learning rate α_k satisfies the Robbins-Monro conditions $\sum_{k=0}^{\infty} \alpha_k = \infty$ and $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$, then the greedy policy with respect to Q converges to the optimal policy with probability 1 as $k \to \infty$.

Proof. Standard Q-learning convergence requires that every state-action pair is updated infinitely often. Our exploring policy π^ϕ prunes actions that are guaranteed to be suboptimal by (9.7), meaning not every state-action pair is updated. However, the key insight is that for every state x, every action that could potentially be optimal is still updated infinitely often (via exploring starts). Thus, while $Q(x, u)$ may not converge to $Q^*(x, u)$ for pruned actions, the greedy policy $\pi(x) \in$

$\arg\max_{u\in\mathcal{U}} Q(x,u)$ converges to the optimal policy $\pi^*(x)$ with probability 1, since the Q-values of all candidate optimal actions either converge to their true values or are dominated by actions that do. □

In the next section, we detail how to select the weighting function β using several heuristics.

Choice of Weighting β

Inspired by acquisition functions in Bayesian optimization [72], we can select efficient weighting functions β. We are interested in improving the best worst-case Q-value $\underline{V}(x)$ at state x. This improvement as a function of the choice of input u is given by

$$I(x,u) = \max\left(0, q - \underline{V}(x)\right), \tag{9.12}$$

in which q denotes the unknown Q-value of input u at state x, which lies between $\underline{Q}(x,u)$ and $\bar{Q}(x,u)$. By assuming that the probability of Q-values within the bounds $\underline{Q}$ and $\bar{Q}$ is uniform, we can calculate the probability that a given input u has an improvement greater than 0. This *probability of improvement* is given by

$$\beta(x,u) = \text{Prob}[I(x,u) > 0] = \frac{\max\left(0, \bar{Q}(x,u) - \underline{V}(x)\right)}{\bar{Q}(x,u) - \underline{Q}(x,u)}. \tag{9.13}$$

Alternatively, we can take the *expected* value of the *improvement* (9.12) to obtain

$$\beta(x,u) = \mathbb{E}[I(x,u)] = \frac{\left(\max\left(0, \bar{Q}(x,u) - \underline{V}(x)\right)\right)^2}{2\left(\bar{Q}(x,u) - \underline{Q}(x,u)\right)}. \tag{9.14}$$

These sampling weights prioritize exploring inputs with a large region of their Q-value uncertainty range higher than the current best-worst-case input. Note that we can assume other probability distributions (with bounded support) for the Q-values between $\underline{Q}$ and $\bar{Q}$. One such example that generalizes the uniform distribution is the Bates distribution, which allows us to assume a higher relative probability concentration around the midpoint of the Q-value bounds.

In the next section, we detail an approach to include data in the method by regularizing the optimization over the model set $\mathcal{P}$ to bias it toward the observed transitions.

9.3.3 Regularization with Observed Data

We propose that by regularizing the optimization over the model set using observed data $\mathcal{D}$, the bounds converge to the optimal Q-function under mild conditions. We

can re-define the Q-bound Bellman updates (9.4) as

$$\begin{aligned}(\underline{\mathcal{T}}\underline{Q})(x,u) &= \inf_{g\in\mathcal{G}} g(x,u) + \gamma \int_{\mathcal{X}} \underline{V}(x')\underline{P}^*(x,u,dx'),\\ (\bar{\mathcal{T}}\bar{Q})(x,u) &= \sup_{g\in\mathcal{G}} g(x,u) + \gamma \int_{\mathcal{X}} \bar{V}(x')\bar{P}^*(x,u,dx'),\end{aligned} \tag{9.15}$$

with

$$\begin{aligned}\underline{P}^*(x,u,dx') &:= \arg\min_{P\in\mathcal{P}} \int_{\mathcal{X}} \underline{V}(x')P(x,u,dx') + \lambda(x,u)d(P,P_E),\\ \bar{P}^*(x,u,dx') &:= \arg\max_{P\in\mathcal{P}} \int_{\mathcal{X}} \bar{V}(x')P(x,u,dx') - \lambda(x,u)d(P,P_E),\end{aligned} \tag{9.16}$$

where we assume that the minimum and maximum exist (which is the case if for instance $\mathcal{P}$ is compact). Here, $P_E : \mathcal{X}\times\mathcal{U}\times\mathcal{B}(\mathcal{X}) \to [0,1]$ denotes the empirical transition kernel obtained from the observed data $\mathcal{D}$, and $\lambda : \mathcal{X}\times\mathcal{U} \to \mathbb{R}_{\geq 0}$ is a regularization parameter. The function $d(P,P_E)$ is a divergence between the transition kernel P and the empirical transition kernel P_E. The divergence is used to bias the optimized transition kernel toward the observed data, while still being within the model set. The function d must satisfy $d(P,P_E) \geq 0$, with equality if and only if $P = P_E$. The variable λ determines the trade-off between the model set and the observed data. It should increase with the size of $\mathcal{D}$ to ensure that the selected model from the set is biased toward the observed data, such that

$$\lim_{T(x,u)\to\infty} \lambda(x,u) = \infty, \tag{9.17}$$

where $T(x,u,x')$ is the number of observed transitions from state x under action u to state x' in the dataset $\mathcal{D}$, and $T(x,u) := \sum_{x'} T(x,u,x')$. In practice, λ can be chosen as any increasing function of the visit count $T(x,u)$ that satisfies (9.17).

Since the rewards are deterministic, whenever a reward r_k is observed at state-action pair (x_k,u_k), the set of reward functions $\mathcal{G}$ can be constrained to include only elements that correctly assign this reward at that state-action pair, i.e.,

$$\mathcal{G} = \{g \in \mathcal{G} \mid g(x_k,u_k) = r_k\}. \tag{9.18}$$

Together with the regularized model set optimization, we obtain a convergence result which is formalized next.

Proposition 9.5. *Assuming finite state-action spaces, let $\lambda : \mathcal{X}\times\mathcal{U} \to \mathbb{R}_{\geq 0}$ be a regularization parameter satisfying (9.17), and let $d(P,P_E)$ be a divergence such that $d(P,P_E) \geq 0$ and $d(P,P_E) = 0$ if and only if $P = P_E$. Let $\mathcal{P}$ and $\mathcal{G}$ be the set of transition kernels and the set of reward functions, respectively. Assume that the reward function is deterministic. If every state-action pair is visited infinitely often, then,*

$$\lim_{N\to\infty} \underline{Q} = \lim_{N\to\infty} \bar{Q} = Q^*. \tag{9.19}$$

Proof. Since every state-action pair is visited infinitely often, $\lambda(x, u) \to \infty$ by (9.17). The regularization term then forces the optimization to select $P = P_E$, as the regularization dominates the objective. With deterministic rewards, $\mathcal{G}$ reduces to the true reward function after one visit per state-action pair through (9.18). By the law of large numbers, P_E converges to the true transition kernel P. As the Bellman operator is a contraction (see [232]), it follows that both the lower and upper Q-function bounds converge to Q^*. □

Note that for any value of λ, the Q-bound updates in (9.4) are still convergent. However, it should be noted that the resulting converged Q-bounds are only guaranteed to bound the optimal Q-function Q^* if $\lambda = 0$ (by Lemma 9.2) or if $P_E = P$ and $\lambda = \infty$ (by Proposition 9.5).

The regularized model set optimization is incorporated into our proposed method, which is summarized in Algorithm 4, where the data-regularized Q-bound iterations are performed every L episodes. To improve learning efficiency, we employ an ϵ-greedy type strategy, balancing exploration/exploitation. Additionally, we clip the Q-function to the bounds whenever they are updated. By combining our previous results, we obtain the following convergence guarantee for the exploring

Algorithm 4 BUMEX: Q-learning with exploring policy and regularized model set optimization

Calculate Q-function bounds $\underline{Q}$ and $\bar{Q}$ using (9.4).
Initialize Q such that $\underline{Q} \leq Q \leq \bar{Q}$.
for Episode $e = 1, 2, 3, \ldots$ **do**
  Initialize state x_0.
  for $k = 0, 1, 2, \ldots$ until terminal **do**
    if rand() $< \epsilon$ **then**
      Select u_k according to exploring policy π^ϕ.
    else
      Select $u_k = \arg\max_{u \in \mathcal{U}} Q(x_k, u)$.
    end if
    Apply u_k and observe r_k and x_{k+1}.
    Reduce the reward function set $\mathcal{G}$ such that $g(x_k, u_k) = r_k$ for all $g \in \mathcal{G}$.
    Update Q using (x_k, u_k, r_k, x_{k+1}) and (9.11).
  end for
  if $e \mod L = 0$ **then**
    Update the Q-bounds using (9.15) with (9.16).
    Clip the Q-function Q to the bounds $\underline{Q}, \bar{Q}$.
  end if
end for

policy.

Theorem 9.6. *Assume finite state and action spaces. Let π^ϕ be the exploring policy defined in (9.10) for which we additionally assume that ζ is positive. Let λ satisfy (9.17). Assume that $d(P, P_E)$ is a divergence with $d(P, P_E) \geq 0$ and $d(P, P_E) = 0$ if and only if $P = P_E$. Assume deterministic rewards, and that every state is visited infinitely often. Then, under Algorithm 4, the Q-bounds converge to the optimal Q-function and the exploring policy π^ϕ converges to the optimal policy with probability 1 as $k \to \infty$.*

Proof. With $\zeta > 0$, the exploring policy assigns positive weight to all actions except those that are certain and guaranteed suboptimal. For uncertain state-action pairs, the policy ensures infinite exploration, causing $\lambda(x, u) \to \infty$ and forcing convergence to the empirical kernel. By Proposition 9.5, the Q-bounds converge to Q^*. The exploring policy then selects only guaranteed optimal actions, converging to the optimal policy. □

9.4 Practical Algorithm: BUMEX

In this section, we consider the finite state and action space case, where the MDP transitions are reduced to a transition probability tensor M, such that

$$\text{Prob}(x_{k+1} = x' \mid x_k = x, u_k = u) = M(x, u, x'), \tag{9.20}$$

with $M \in [0,1]^{|\mathcal{X}|\times|\mathcal{U}|\times|\mathcal{X}|}$ and $\sum_{x'\in\mathcal{X}} M(x, u, x') = 1$ for all $(x, u) \in \mathcal{X} \times \mathcal{U}$. The MDP model set $\hat{\mathcal{M}}$ is defined as a bounded-parameter MDP model [228]. The set that the true transition kernel M belongs to is defined as

$$\mathcal{P} = \left\{ M \;\middle|\; \underline{M} \leq M \leq \bar{M}, \sum_{x'\in\mathcal{X}} M(x, u, x') = 1, \quad \forall\, x, u \right\}, \tag{9.21}$$

for known, well-defined lower and upper bounds $\underline{M}, \bar{M} \in [0,1]^{|\mathcal{X}|\times|\mathcal{U}|\times|\mathcal{X}|}$. The reward function set that the true reward function g belongs to is defined as $\mathcal{G} = \{g \mid \underline{g} \leq g \leq \bar{g}, \quad \forall\, x, u\}$. Thus, we assume that the true transition kernel and reward function are bounded by known upper and lower bounds.

The function d is chosen to be the Kullback-Leibler (KL) divergence between the transition kernel and the observed data, given by

$$d(M, M_E) = \sum_{x'\in\mathcal{X}} M(x, u, x') \log\left(\frac{M(x, u, x')}{M_E(x, u, x')}\right), \tag{9.22}$$

where M_E is the empirical transition probability tensor from the observed data $\mathcal{D}$ through $M_E(x, u, x') = \frac{T(x,u,x')}{T(x,u)}$. We choose the KL-divergence since it satisfies

$d(P, P_E) \geq 0$ with equality if and only if $P = P_E$, is convex, and is a natural choice for comparing probability distributions.

The resulting Bellman updates are given by

$$\begin{aligned}(\underline{\mathcal{T}}\underline{Q})(x,u) &= \underline{g}(x,u) + \gamma \sum_{x' \in \mathcal{X}} \underline{V}(x')\underline{M}^*(x,u,x'),\\ (\bar{\mathcal{T}}\bar{Q})(x,u) &= \bar{g}(x,u) + \gamma \sum_{x' \in \mathcal{X}} \bar{V}(x')\bar{M}^*(x,u,x'),\end{aligned} \tag{9.23}$$

with the regularized model optimization given by

$$\begin{aligned}\underline{M}^*(x,u,x') &= \arg\min_{M \in \mathcal{P}} \sum_{x' \in \mathcal{X}} \underline{V}(x')M(x,u,x') + \lambda d(M, M_E),\\ \bar{M}^*(x,u,x') &= \arg\max_{M \in \mathcal{P}} \sum_{x' \in \mathcal{X}} \bar{V}(x')M(x,u,x') - \lambda d(M, M_E).\end{aligned} \tag{9.24}$$

We propose to select $\lambda = 0$ in the absence of data ($T(x,u) = 0$) and otherwise as a function of system properties and the number of observed transitions from every state-action pair, such that

$$\lambda(x,u) = c\sqrt{\frac{T(x,u)}{\log\left(|\mathcal{X}||\mathcal{U}|/\delta\right)}}, \tag{9.25}$$

where c is a constant, and δ is a confidence parameter. This formulation is based on Hoeffding's inequality [233]. This usage is similar to the PAC-MDP framework [234], as implemented, e.g., in the UCRL2 algorithm [225].

The algorithm presented in this section is a straightforward extension of the tabular Q-learning algorithm, where we can use the Q-function bounds to guide the agent's exploration. The Q-bound updates are computationally efficient: without regularization, they reduce to simple min/max operations over the model set bounds, while with regularization, the optimization remains convex and tractable. The computational complexity scales similarly to other finite MDP methods like UCRL2 or PSRL, with runtime dominated by the state-action space size rather than the model set optimization. Note that all the convergence properties of Q-learning discussed in the previous sections are retained. In some cases, we can even guarantee finite-time convergence of the exploring policy to the optimal policy, as shown in the next section.

9.4.1 Finite-Time Convergence

We can guarantee finite-time convergence of the proposed exploration strategy under the following conditions:

Assumption 9.7. For every $(x,u) \in \mathcal{X} \times \mathcal{U}$, one of the following holds:

1. **Deterministic transitions:** For all $x' \in \mathcal{X}$, $M(x, u, x') \in \{0, 1\}$.
2. **Tight bounds:** For all $x' \in \mathcal{X}$, $\underline{M}(x, u, x') = \bar{M}(x, u, x')$.

Furthermore, $\lambda(x, u)$ is set to rely entirely on the data for the deterministic case, that is, $\lambda(x, u) = \infty$ if 1) holds and $T(x, u) > 0$.

The finite-time convergence result is formalized in the following lemma.

Lemma 9.8. *Let π^ϕ be the exploring policy defined in* (9.10) *and let Q be the Q-function from Algorithm 4, using model set $\hat{\mathcal{M}}$ and divergence d with $d(P, P_E) \geq 0$ and $d(P, P_E) = 0$ if and only if $P = P_E$. Assume deterministic rewards. Assume that every state is visited infinitely often (e.g., via exploring starts), and that Assumption 9.7 holds. Then, the Q-bounds converge to Q^* in finite time with probability 1, and the π^ϕ converges to the optimal policy in finite time.*

Proof. Consider any state-action pair (x, u). The exploring policy ensures that any state-action pair with uncertain Q-bounds is eventually sampled at least once (in finite time). Under Assumption 9.7, we have:

1. In the *deterministic case*, a finite number of visits yields the exact empirical kernel $P_E(x, u, \cdot)$; with $\lambda(x, u) = \infty$, the optimization in (9.16) forces $P = P_E$, making the Q-bounds exact in finite time.
2. In the *tight-bound case*, the Q-bound update reduces to the standard Bellman update, since the model set contains only a single model.

The reward function set $\mathcal{G}$ is reduced to the true reward function g in finite time, since we have finite state and action spaces. The model uncertainty is thus completely eliminated. By Theorem 9.6, the exploring policy therefore converges to the optimal policy in finite time with probability 1. □

Although this chapter aims to present a general theoretical framework that can be cast into both finite and infinite state/action spaces, formulating a practical algorithm for the infinite-state, infinite-action case is considerably more challenging. In practice, one must resort to function approximation techniques to parameterize the Q-function. Such a parameterization typically leads to difficulties when performing optimistic max-max optimization because the overestimation inherent in the optimistic operator can result in divergence issues or unstable learning dynamics [235]. The development of such an algorithm is left for future work.

A practical workaround is to construct a finite abstraction of the original system. By discretizing the state and action spaces, one can apply the proposed exploration strategy and Q-bound updates in a tractable manner, with approximation accuracy improving as the discretization becomes finer. This approach faces the same computational scaling challenges as other state-of-the-art finite MDP RL methods such as the ones used in the comparison in the next section.

9.5 Results

We apply the proposed practical algorithm to three benchmark reinforcement learning environments: Frozen Lake, Cartpole, and Taxi from the OpenAI Gym [236].

9.5.1 Frozen Lake Environment

In the Frozen Lake environment, the agent navigates a gridworld to avoid obstacles and reach a goal state, receiving a reward of 1 upon success and 0 otherwise. The transitions are stochastic due to the slippery nature of the ice, with the agent moving in either the intended direction or one of the two perpendicular directions, each with probability 1/3.

We define the model set to include knowledge about adjacency in the gridworld. In particular, we assume that all transitions for which the true model has probability zero are known, while all others have probabilities between 0 and 1. The reward function set $\mathcal{G}$ contains only the true reward function g, assuming full knowledge of the rewards. This model set definition is similar to what a human player might assume when playing the game, i.e., the agent knows the layout of the gridworld and the locations of obstacles and goals, but not the exact transition probabilities. The results are shown in Figure 9.2.

We compare standard (randomly exploring) ϵ-greedy Q-learning, Q-learning with UCB1 exploration [224], UCRL2 [225], and PSRL [226] to two variants of BUMEX: one without regularization ($L = \infty$) and one with regularization ($L = 100$). All experiments use $\xi = 1$, $\zeta = 0$, β as in (9.14), $c = 5$, $\delta = 0.05$, $\alpha = 0.05$, and $\gamma = 0.95$, with ϵ decaying exponentially from 1 to 0.01. The results indicate that BUMEX converges to the optimal policy quickly and consistently, even when the Q-bound iterations are only performed at the beginning. When performing Q-bound iterations (with regularization) every 100 episodes, BUMEX converges even faster, although this comes at an additional computational cost induced by the repeated calculation of Bellman operators. Note that methods such as UCRL2 and PSRL also rely on repeated Bellman operations with comparable computational costs.

9.5.2 Cartpole Environment

In the Cartpole environment, the agent is tasked with balancing a pole on a cart by moving the cart left or right, receiving a reward of 1 per time step the pole is balanced. The transitions are deterministic and governed by the physics of the cartpole system. We create finite state and action spaces by discretizing the continuous state and action spaces, which introduces stochasticity into the transition model. Furthermore, we perform quadratic reward shaping during the training phase to speed up learning.

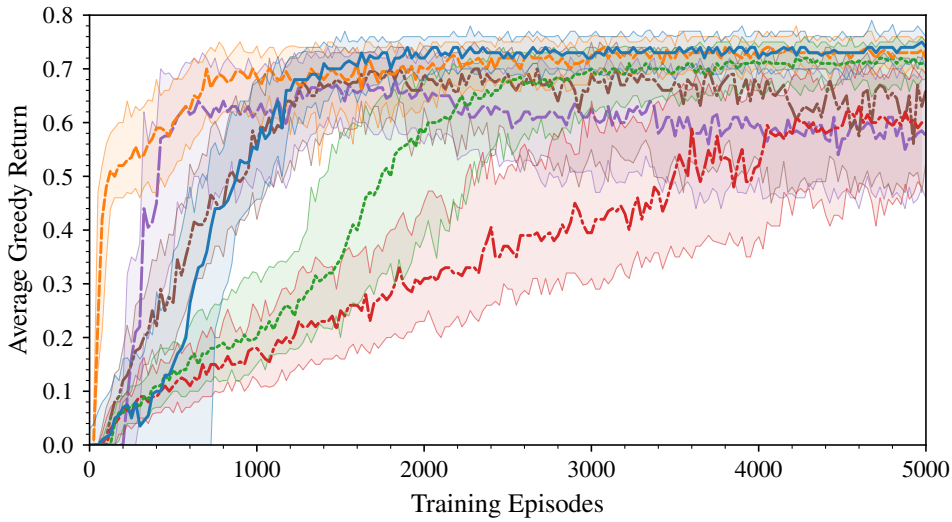


Figure 9.2. Evaluation performance of BUMEX with $L = \infty$ (——) and $L = 100$ (- - -) versus standard ϵ-greedy Q-learning (····), Q-learning with UCB exploration (-·-·), UCRL2 (- - -), and PSRL (-··-) in the Frozen Lake environment. Plotted are the median, and the 25th and 75th percentiles of 100 Monte Carlo runs, where the performance is evaluated every 25 training episodes by averaging the returns obtained by applying the greedy policy for 100 episodes.

We define the model set to include knowledge about the cartpole dynamics, i.e., the cartpole is governed by Newtonian physics. In particular, we assume that the dynamics equations are known up to an unknown mass parameter. We obtain the set $\mathcal{P}$ by looping over the set of possible mass values, obtaining the transition probability tensor for each, then taking the max and min over these tensors to obtain $\bar{M}$ and $\underline{M}$ in (9.21). The reward function set $\mathcal{G}$ contains only the true reward function g, i.e., we assume full knowledge of the rewards. The results are shown in Figure 9.3.

We compare the algorithms ϵ-greedy, UCB1, PSRL, BUMEX with $L = \infty$ and $L = 200$, with the same ξ, ζ, β, c, and δ as in Section 9.5.1 (UCRL2 is excluded due to poor performance). We use $\alpha = 0.03$, $\gamma = 0.97$, and ϵ decaying exponentially from 1 to 0.001. The results are similar to those in the Frozen Lake environment, showing that the regularized method converges fastest and most consistently, followed by the non-regularized version. The more limited performance gain from regularization in this environment can be attributed to the larger state space dimension.

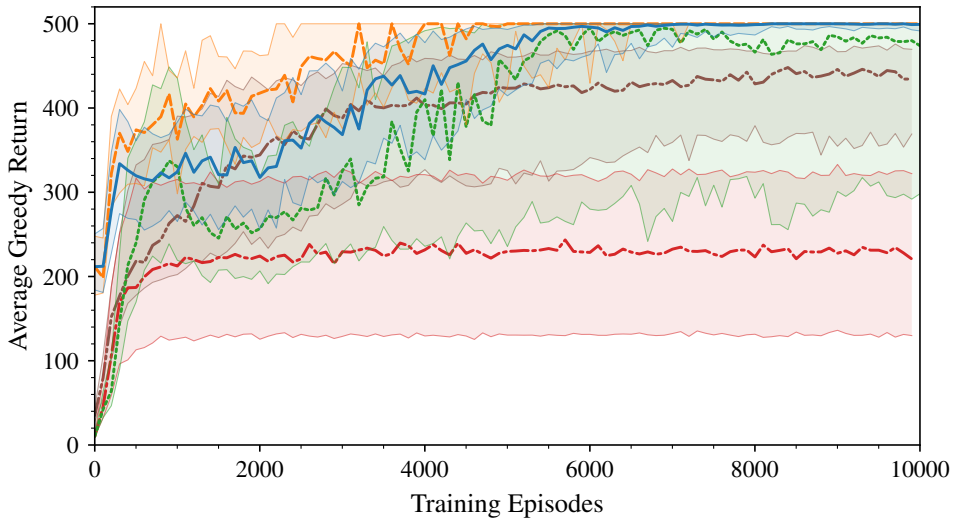


Figure 9.3. Evaluation performance of BUMEX with $L = \infty$ (——) and $L = 200$ (- - -) versus standard ϵ-greedy Q-learning (····), Q-learning with UCB exploration (-·-·-), and PSRL (-··-··-) in the Cartpole environment. Plotted are the median, and the 25th and 75th percentiles of 100 Monte Carlo runs, where the performance is evaluated every 100 training episodes by averaging the returns obtained by applying the greedy policy for 100 episodes.

9.5.3 Taxi Environment

The Taxi environment consists of a more complicated gridworld, in which the agent is tasked with picking up and dropping off passengers at dedicated locations. The agent is rewarded -1 at every timestep, -10 for illegal passenger pickups/dropoffs, and +20 for a successful passenger dropoff. The gridworld contains walls which the agent cannot move through. Note that this environment is deterministic.

Our prior model knowledge largely takes the same form as in Section 9.5.1, i.e., we assume that the agent knows the adjacency structure of the gridworld and the locations of the pickup and dropoff points. We assume the agent does not have knowledge of wall placements or the reward function structure.

We compare the same algorithms (ϵ-greedy, UCB1, UCRL2, PSRL, BUMEX with $L = \infty$ and $L = 100$) as in the previous sections, with the same ξ, ζ, β, c, and δ. We use $\alpha = 0.03$, $\gamma = 0.98$, and ϵ decaying exponentially from 1 to 0.01. The results are shown in Figure 9.4 and are consistent with the other two environments. This experiment specifically highlights the benefits of having accurate dynamics knowledge while learning unknown rewards. The proposed method converges very quickly once the deterministic reward function has been learned.

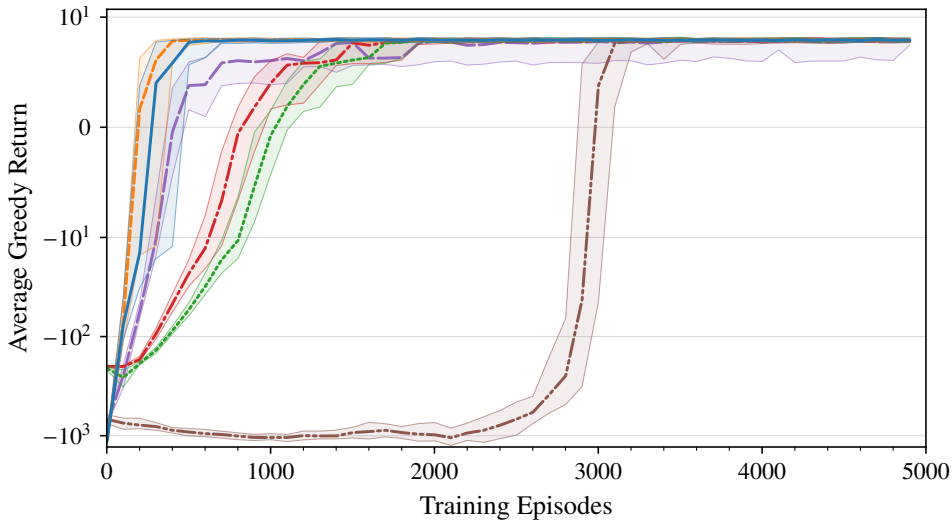


Figure 9.4. Evaluation performance of BUMEX with $L = \infty$ (——) and $L = 100$ (- - -) versus standard ϵ-greedy Q-learning (····), Q-learning with UCB exploration (-·-·-), UCRL2 (- - -), and PSRL (-··-··-) in the Taxi environment. Plotted are the median, and the 25th and 75th percentiles of 100 Monte Carlo runs, where the performance is evaluated every 100 training episodes by averaging the returns obtained by applying the greedy policy for 100 episodes.

9.6 Conclusions and Future Work

In this chapter, we proposed an exploration strategy for reinforcement learning that leverages model-based Q-function bounds to guide the agent's exploration. Our work establishes multiple theoretical results that guarantee the convergence of the proposed exploration strategy to the optimal policy in a general MDP setting. Furthermore, in the finite state and action space case, and under reasonable assumptions on the model set, we obtain a practical algorithm, BUMEX, with the same convergence guarantees. We also demonstrated that, under additional assumptions on the transition probabilities, our method achieves finite-time convergence to the optimal policy. The effectiveness of the proposed strategy was validated in several Gym environments, where it outperformed standard exploration methods.

Future work includes extending the theoretical results to infinite state-action spaces, e.g., through function approximation. Additionally, a sample complexity analysis of BUMEX is desired to quantify the efficiency compared to state-of-the-art RL methods.

10 Conclusions

Given noisy observations, what can we learn about the system that produced them? Each chapter of this thesis answered a version of this question for a different problem, and read on their own they have little in common. They range from calibrating optical aberrations in STEM and TEM, to nanometer-precision motion control, to data-efficient reinforcement learning. However, when we step back, these chapters share an overarching philosophy on learning from limited and indirect measurements. This concluding chapter first states what the thesis achieved for electron microscopy, then presents the themes shared across the chapters, and finally returns to the question the introduction raised about the role of structure in the learning context.

10.1 Toward Autonomous Electron Microscopy

The introduction set out four challenges for electron microscopy. The first, the correction and calibration of aberrations (C1), runs through the first two parts of the thesis. Chapter 2 gives a fully data-driven calibration for the scanning transmission electron microscope (STEM). A neural network maps each Ronchigram to a scalar measure of the distance to a calibrated state, and a Bayesian optimization loop with a problem-specific acquisition function drives that measure down, calibrating defocus and two-fold astigmatism in roughly twenty Ronchigrams acquired in under fifteen seconds. The scalar measure is also the main limitation of this approach, since it loses information as the number of aberrations grows. Chapter 3

removes that limitation. A variational autoencoder learns a multi-dimensional latent representation of the Ronchigram, and an expectation-maximization procedure estimates the aberration state together with the mismatch between simulated and real images. The evenness of the Ronchigram power spectrum in the aberration space makes this joint estimate identifiable, and the resulting calibration more than halves the error of existing automated methods while converging in tens of seconds.

Chapter 4 carries C1 into the transmission electron microscope (TEM), where the aberrations are estimated from the image shifts that beam tilts induce. The core difficulty is that the speed and accuracy of the estimate depend on which tilts are applied. Casting the choice of tilts as a sensor-scheduling problem and selecting them to minimize the trace of the predicted estimation covariance reaches a target accuracy in about forty-two percent fewer tilts than an uninformed pattern, and experiments on a real TEM reproduce the predicted covariances. The method also handles polycrystalline specimens, which the classical Zemlin tableau cannot.

The third challenge, three-dimensional imaging and specimen positioning (C3), is the subject of Chapter 5. Electron tomography reconstructs a specimen from a series of images taken at different tilt angles, and the specimen must stay fixed at the point of interest as the stage tilts. The stage is a four-degree-of-freedom piezo-stepper whose actuators suffer from hysteresis and whose position cannot be measured directly at the point of interest. A data-driven feedforward controller compensates the hysteresis and the repeatable positioning error in the commutation-angle domain through iterative learning control, while the microscope images themselves provide the point-of-interest position measurement that drives the ILC learning. Together these hold the specimen in place to within nanometers on an operational microscope.

The fourth challenge, throughput, automation, and the data challenge (C4), runs through all the applied chapters. Each calibration and positioning method replaces expert operator time with an automated procedure that raises throughput, and each interprets high-dimensional image data directly, which answers the data side of the same challenge.

Together, these methods remove parts of the manual work that stands between current instruments and autonomous operation. Each was developed for a specific task. However, the ideas behind them recur across tasks, and the next section turns to those.

10.2 Lessons Learned

Most of the chapters in this thesis attempt to solve inverse problems, which consist of recovering a quantity of interest from indirect observations through a forward model. A recurring difficulty in such problems is identifiability: distinct values of the quantity can produce the same observations, so the data alone cannot decide

between them. Structural assumptions are how we can restore a unique solution. Many of the estimation chapters can be cast this way, since they rest on Bayesian inference, with the Kalman filter of Chapter 4 and the Gaussian process regression of Chapters 2 and 7 as two forms of it (Chapter 7 even makes this connection explicit by unifying the two forms). The cases worth examining in detail are those where identifiability is genuinely at stake.

Optical aberration estimation is the clearest example. Its forward map is non-injective, since distinct aberration states can produce visually indistinguishable Ronchigrams. Overfocus and underfocus, for instance, both appear as a defocused pattern, so a single observation cannot be used to uniquely determine the aberration state. Chapter 2 and Chapter 4 resolve this through the history of observations and a rich choice of inputs. Chapter 2 compresses the Ronchigram to a scalar distance from the calibrated state, turning estimation into a search for the input that minimizes this distance. Bayesian optimization is well suited to that search. It makes large, informative jumps across the parameter space, and it conditions on the full history of observations while searching in the parameter space, which is what separates overfocus from underfocus. Chapter 4 achieves the same effect by estimating the aberrations based on the full history of tilts and the resulting image shifts.

Chapter 3 faces a harder version of the same problem. There the estimate must include both a proxy for the aberration state and the mismatch between the simulator and the real instrument, and in general a continuum of state-and-mismatch pairs explains the same observations. The joint problem is therefore not identifiable. The chapter shows that further structural assumptions remove the ambiguity. The evenness of the Ronchigram power spectrum in the aberration space constrains the Gaussian process model of the mismatch and makes the joint estimate identifiable, provided the history of measurements is rich enough.

The same concern reappears in a different guise in Chapters 3 and 6, which both use variational autoencoders (VAEs). A VAE fits a forward model and an inverse model at once, with the latent space as the hidden quantity, and many latent spaces explain the data equally well. Chapter 6 constrains this freedom in two ways. It prescribes the topology of the latent space by building the required product or quotient structure into the encoder and decoder, where a quotient is a symmetry reduction of the latent space. It also adds anchor constraints that pin chosen observations to fixed latent locations, which ties the coordinate system to reality. The symmetry used here connects to Chapter 3, though it enters differently. Chapter 6 builds the symmetry into the latent space through the autoencoder, whereas Chapter 3 reads the symmetry off the data and uses it to constrain the forward model that maps the aberration state to the latent representation.

A related idea is the explicit closing of the gap between a model and reality. The simulation-to-reality gap of P2 is closed directly in Chapter 3, which estimates

the mismatch from a small amount of real data. Chapter 9 does something similar for reinforcement learning, where its data-based regularization pulls the model set toward the observed transitions, bringing the assumed dynamics closer to the true system as data accumulates.

Several chapters use a model of the remaining uncertainty to choose what to measure or do next. Chapter 4 chooses each beam tilt by minimizing the trace of the predicted estimation covariance (A-optimality), so that the next tilt is the one expected to reduce the uncertainty the most. Chapter 2 selects each corrector setting with a custom acquisition function evaluated on the Gaussian process posterior, weighing each candidate by its distance from the current best estimate and by the probability that it improves on it, so the setting tested next is the one whose outcome would most change the calibration if the current estimate were wrong. The same idea appears in Chapter 9, which adapts acquisition functions from Bayesian optimization to reinforcement learning and uses bounds on the Q-function to select the next action and prune the ones that are provably suboptimal. In each case, choosing the measurement or action expected to be most informative is the experiment-design counterpart of the Fisher information from the introduction.

An additional idea concerns the choice of modeling assumptions. Across the chapters, the assumptions are chosen strong enough to make the problem tractable and weak enough to stay accurate. Smoothness, assumed in the Gaussian process models of Chapters 2 and 7, is a mild assumption that admits flexible methods such as Bayesian optimization, at the cost of efficiency. Convexity, used in the quadratic Q-function of Chapter 8 and the model-set optimization of Chapter 9, is far more restrictive but admits efficient solvers. The same trade-off appears in the separable kernel that reduces Chapter 7 to a finite Kalman filter. Where an assumption is known to be inexact, the chapters quantify the cost. Chapter 4 shows empirically that its physical model is accurate enough for the estimator to perform well, and Chapter 7 bounds the error introduced by reducing the model order.

10.3 Structure vs. Scaling

The introduction raised the Bitter Lesson as an objection to the premise of this thesis [35]. It argues that across the history of artificial intelligence, general methods that scale with computation have outperformed methods built on human-engineered domain knowledge. Much of this thesis is, in fact, built from general-purpose tools: convolutional neural networks, Gaussian processes, Bayesian optimization, Kalman filters, Q-learning, and gradient-based and convex optimization. These are not curated domain knowledge, and to that extent the work agrees with the Bitter Lesson. The structural assumptions sit on top of these tools, and the question is when they are necessary rather than optional.

There are two regimes where structure is necessary, and they are distinct. The

first is identifiability. When the goal is to recover a quantity of interest and the forward model is non-injective, no quantity of data resolves it without a structural prior, regardless of scale. This matters only when the quantity itself is the goal. For problems where only a prediction or a decision is needed, two parameter values that produce the same observations are equally acceptable, and identifiability can be ignored. The joint estimation in Chapter 3 is a case where it cannot be ignored, since we can only calibrate the optical aberrations if they are identifiable from the data.

The second regime is the combination of bounded real data (P1) and a simulator that does not match the real instrument (P2). Either condition alone leaves scaling a way out. With abundant real data, a model can be trained on the instrument directly. With a simulator that matches reality, more simulated data closes the gap. When both hold at once, neither remedy is available, and improving data-efficiency is the only option. Chapter 3 shows this concretely: for the same number of Ronchigram observations, its structured estimate more than halves the error of the simplified approach of Chapter 2.

Where neither regime applies, the Bitter Lesson likely holds, and the effort of building in structure is better spent on scaling. This thesis marks where that view breaks down rather than refuting it. Electron microscopy sits in the regime where it does: the data is bounded, the simulator is approximate, and we are dealing with non-injective forward maps.

10.4 Limitations

The thesis intentionally focuses on specific aspects of electron microscopy. Scanning electron microscopy and specimen preparation are considered out of scope from the start. Dose-limited imaging (C2), a central motivation for the field, is addressed only indirectly, since methods that need fewer Ronchigrams or fewer tilts spend less dose, while the dose-limited regime itself is left untreated. The fully autonomous operation of C4 is brought closer but not reached.

The structure-first approach behind these contributions also has limits of its own. As the previous section discussed, it helps where identifiability or the combination of P1 and P2 is at stake, and where neither holds, its advantage over collecting more data is small. It further depends on the structure being known and correct, since a wrong model or symmetry pulls the solution toward the wrong answer, and a problem with no usable structure leaves nothing to build in.

A second kind of limitation concerns maturity. The methods of Parts I and II are validated on real instruments and built with practical use in mind, but they are not yet deployed, which would take further integration and engineering. The methods of Part III are demonstrated mostly on simulated case studies of modest size, so their performance on large-scale real datasets is open. The chapters analyze

the computational complexity of these methods and suggest how to scale them, for instance through function approximation, but that scaling has not yet been carried out in practice.

10.5 Recommendations

The clearest direction for future work is fully autonomous electron microscopy, in particular STEM calibration and continuous-tilt TEM tomography. For STEM calibration, the main remaining challenge is scaling to higher-order aberrations. Chapter 3 treats defocus and two-fold astigmatism, but higher-order aberrations have more localized effects on the Ronchigram, and thus it is recommended to divide the Ronchigrams into patches with local Fourier features extracted from each. The estimation problem also grows rapidly in dimension, and therefore a hierarchical approach that resolves lower-order aberrations first and then refines higher-order terms would keep the dimensionality tractable. A further open problem is corrector fidelity. The calibration loop assumes that the correctors apply exactly the commanded changes, but this relation carries uncertainty in practice and is not currently accounted for in the estimation.

For specimen positioning, the remaining gap is full three-dimensional POI tracking. The ILC framework of Chapter 5 tracks the specimen's in-plane position from image cross-correlation and encoder measurements, but has no measurement of the axial coordinate (the specimen's displacement along the beam). That axial displacement determines the defocus seen in each TEM image. The tilt-based aberration estimation of Chapter 4 estimates defocus quickly and without requiring a particular specimen type, so it could in the future supply the out-of-plane position measurement that Chapter 5 currently lacks. Fusing all three sources (encoder readings, image-based in-plane tracking, and tilt-derived defocus) into a single high-rate 3D POI estimate would benefit both the feedforward learning of Chapter 5 and the closed-loop feedback.

For the general methods of Part III, several concrete extensions remain open. The latent topology of Chapter 6 could be learned from data rather than prescribed by hand. The Dynamic Gaussian Process of Chapter 7 could be given control inputs and the posterior-covariance-based sensor scheduling that it already points to. The two convex methods of Chapter 8 invite a convergence analysis and conditions under which the relaxation is exact. The bounded-uncertainty exploration of Chapter 9 could be extended to continuous state-action spaces through function approximation. Each of these would widen the range of problems the methods can reach.

10.6 Closing Remarks

This thesis set out to learn from limited and indirect measurements, with electron microscopy as the case that motivated the question. The answer it offers is that structure is what makes such learning possible: it determines which quantities can be identified, it sets how much each measurement reveals, and it allows us to bridge the simulation-to-reality gap rather than ignore it.

The history of electron microscopy is one of instruments that grow more capable and, with that, harder to operate. The methods in this thesis let learning-based algorithms take on part of that burden, enabling calibration and precision positioning based on a handful of measurements. As this kind of automation matures, the operator's effort can move from tuning the microscope toward the science it serves. The same reasoning applies wherever real data is scarce, and a simulator stands in for reality. More fundamentally, it applies wherever observations alone cannot uniquely identify what produced them. What noisy observations can teach us, it turns out, depends entirely on what we already know.

Bibliography

[1] G. Davies, "Ultra-High-Quality Imaging and Analysis, All in One Tool," 2019.

[2] M. Knoll and E. Ruska, "Das Elektronenmikroskop," *Zeitschrift für Physik*, vol. 78, no. 5-6, pp. 318–339, may 1932.

[3] O. Scherzer, "Über einige Fehler von Elektronenlinsen," *Zeitschrift für Physik*, vol. 101, no. 9-10, pp. 593–603, sep 1936.

[4] A. V. Crewe, J. Wall, and J. Langmore, "Visibility of Single Atoms," *Science*, vol. 168, no. 3937, pp. 1338–1340, jun 1970.

[5] M. Haider, S. Uhlemann, E. Schwan, H. Rose, B. Kabius, and K. Urban, "Electron microscopy image enhanced," *Nature*, vol. 392, no. 6678, pp. 768–769, apr 1998.

[6] O. L. Krivanek, N. Dellby, A. J. Spence, R. A. Camps, and L. M. Brown, "Aberration Correction in the STEM," in *Electron Microscopy and Analysis*, J. Rodenburg, Ed., vol. 153. IOP Publishing, 1997, pp. 17–22.

[7] O. Krivanek, N. Dellby, and A. Lupini, "Towards sub-Å electron beams," *Ultramicroscopy*, vol. 78, no. 1-4, pp. 1–11, jun 1999.

[8] P. E. Batson, N. Dellby, and O. L. Krivanek, *Sub-ångstrom resolution using aberration corrected electron optics*. Nature Publishing Group, aug 2002, vol. 418, no. 6898.

[9] J. Dubochet, M. Adrian, J.-J. Chang, J.-C. Homo, J. Lepault, A. W. McDowall, and P. Schultz, "Cryo-electron microscopy of vitrified specimens," *Quarterly Reviews of Biophysics*, vol. 21, no. 2, pp. 129–228, may 1988.

[10] J. Frank, *Three-Dimensional Electron Microscopy of Macromolecular Assemblies*, 2nd ed. Oxford University Press, 2006.

[11] R. Henderson, "The potential and limitations of neutrons, electrons and X-rays for atomic resolution microscopy of unstained biological molecules," *Quarterly Reviews of Biophysics*, vol. 28, no. 2, pp. 171–193, may 1995.

[12] D. Wrapp, N. Wang, K. S. Corbett, J. A. Goldsmith, C.-L. Hsieh, O. Abiona, B. S. Graham, and J. S. McLellan, "Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation," *Science*, vol. 367, no. 6483, pp. 1260–1263, mar 2020.

[13] W. Kühlbrandt, "The Resolution Revolution," *Science*, vol. 343, no. 6178, pp. 1443–1444, mar 2014.

[14] J. Barthel and A. Thust, "On the optical stability of high-resolution transmission electron microscopes," *Ultramicroscopy*, vol. 134, pp. 6–17, nov 2013.

[15] F. Zemlin, K. Weiss, P. Schiske, W. Kunath, and K. H. Herrmann, "Coma-free alignment of high resolution electron microscopes with the aid of optical diffractograms," *Ultramicroscopy*, vol. 3, no. C, pp. 49–60, jan 1978.

[16] C. Ophus, H. I. Rasool, M. Linck, A. Zettl, and J. Ciston, "Automatic software correction of residual aberrations in reconstructed HRTEM exit waves of crystalline samples," *Advanced Structural and Chemical Imaging*, vol. 2, no. 1, p. 15, dec 2016.

[17] J. Barthel and A. Thust, "Aberration measurement in HRTEM: Implementation and diagnostic use of numerical procedures for the highly precise recognition of diffractogram patterns," *Ultramicroscopy*, vol. 111, no. 1, pp. 27–46, dec 2010.

[18] P. Hawkes, "The correction of electron lens aberrations," *Ultramicroscopy*, vol. 156, pp. A1–A64, sep 2015.

[19] R. Egerton, "Radiation damage to organic and inorganic specimens in the TEM," *Micron*, vol. 119, no. November 2018, pp. 72–87, apr 2019.

[20] R. M. Glaeser, "How good can cryo-EM become?" *Nature Methods*, vol. 13, no. 1, pp. 28–32, jan 2016.

[21] S. Chen and P. Gao, "Challenges, myths, and opportunities of electron microscopy on halide perovskites," *Journal of Applied Physics*, vol. 128, no. 1, jul 2020.

[22] P. Midgley and M. Weyland, "3D electron microscopy in the physical sciences: the development of Z-contrast and EFTEM tomography," *Ultramicroscopy*, vol. 96, no. 3-4, pp. 413–431, sep 2003.

[23] E. Nogales and J. Mahamid, "Bridging structural and cell biology with cryo-electron microscopy," *Nature*, vol. 628, no. 8006, pp. 47–56, apr 2024.

[24] S. Devasia, E. Eleftheriou, and S. O. R. Moheimani, "A Survey of Control Issues in Nanopositioning," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 5, pp. 802–823, sep 2007.

[25] D. N. Mastronarde, "Automated electron microscope tomography using robust prediction of specimen movements," *Journal of Structural Biology*, vol. 152, no. 1, pp. 36–51, 2005.

[26] S. H. Scheres, "RELION: Implementation of a Bayesian approach to cryo-EM structure determination," *Journal of Structural Biology*, vol. 180, no. 3, pp. 519–530, dec 2012.

[27] A. Punjani, J. L. Rubinstein, D. J. Fleet, and M. A. Brubaker, "cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination," *Nature Methods*, vol. 14, no. 3, pp. 290–296, mar 2017.

[28] A. Tejada, A. J. den Dekker, and W. Van den Broek, "Introducing measure-by-wire, the systematic use of systems and control theory in transmission electron microscopy," *Ultramicroscopy*, vol. 111, no. 11, pp. 1581–1591, nov 2011.

[29] S. R. Spurgeon, C. Ophus, L. Jones, A. Petford-Long, S. V. Kalinin, M. J. Olszta, R. E. Dunin-Borkowski, N. Salmon, K. Hattar, W.-C. D. Yang, R. Sharma, Y. Du, A. Chiaramonti, H. Zheng, E. C. Buck, L. Kovarik, R. L. Penn, D. Li, X. Zhang, M. Murayama, and M. L. Taheri, "Towards data-driven next-generation transmission electron microscopy," *Nature Materials*, vol. 20, no. 3, pp. 274–279, mar 2021.

[30] S. V. Kalinin, C. Ophus, P. M. Voyles, R. Erni, D. Kepaptsoglou, V. Grillo, A. R. Lupini, M. P. Oxley, E. Schwenker, M. K. Y. Chan, J. Etheridge, X. Li, G. G. D. Han, M. Ziatdinov, N. Shibata, and S. J. Pennycook, "Machine learning in scanning transmission electron microscopy," *Nature Reviews Methods Primers*, vol. 2, no. 1, p. 11, mar 2022.

[31] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM, 2005.

[32] E. T. Jaynes, *Probability Theory: The Logic of Science*, G. L. Bretthorst, Ed. Cambridge University Press, apr 2003.

[33] J. L. Doob, "Application of the theory of martingales," in *Le Calcul des Probabilités et ses Applications*, ser. Colloques Internationaux du Centre National de la Recherche Scientifique, vol. 13. Paris: CNRS, 1949, pp. 22–28.

[34] H. Cramér, *Mathematical Methods of Statistics*. Princeton, NJ: Princeton University Press, 1946.

[35] R. Sutton, "The Bitter Lesson," 2019.

[36] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling Laws for Neural Language Models," jan 2020.

[37] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre, "Training Compute-Optimal Large Language Models," *Advances in Neural Information Processing Systems*, vol. 35, no. 2020, pp. 1–36, mar 2022.

[38] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, dec 2018.

[39] V. Udandarao, A. Prabhu, A. Ghosh, Y. Sharma, P. Torr, A. Bibi, S. Albanie, and M. Bethge, "No "Zero-Shot" Without Exponential Data: Pretraining Concept Frequency Determines Multimodal Model Performance," in *Advances in Neural Information Processing Systems 37*, vol. 37, no. NeurIPS. San Diego, California, USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2024, pp. 61 735–61 792.

[40] D. Silver and R. S. Sutton, "Welcome to the Era of Experience," in *Designing an Intelligence*. MIT Press, 2025, ch. pre-print.

[41] J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, M. M. A. Patwary, Y. Yang, and Y. Zhou, "Deep Learning Scaling is Predictable, Empirically," pp. 1–19, dec 2017.

[42] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era," in *2017 IEEE International Conference on Computer Vision (ICCV)*, vol. 2017-Octob. IEEE, oct 2017, pp. 843–852.

[43] B. Sorscher, R. Geirhos, S. Shekhar, S. Ganguli, and A. S. Morcos, "Beyond neural scaling laws: beating power law scaling via data pruning," *Advances in Neural Information Processing Systems*, vol. 35, no. NeurIPS 2022, apr 2023.

[44] M. Vulović, E. Franken, R. B. Ravelli, L. J. van Vliet, and B. Rieger, "Precise and unbiased estimation of astigmatism and defocus in transmission electron microscopy," *Ultramicroscopy*, vol. 116, pp. 115–134, may 2012.

[45] M. Rudnaya, "Automated focusing and astigmatism correction in electron microscopy," PhD thesis, Eindhoven University of Technology, Eindhoven, 2011.

[46] Thermo Fisher Scientific, "Sherpa User Manual," 2019.

[47] Corrected Electron Optical Systems (CEOS GmbH), "STEM Aberration Correctors - DCOR/ASCOR."

[48] G. Bertoni, E. Rotunno, D. Marsmans, P. Tiemeijer, A. H. Tavabi, R. E. Dunin-Borkowski, and V. Grillo, "Near-real-time diagnosis of electron optical phase aberrations in scanning transmission electron microscopy using an artificial neural network," *Ultramicroscopy*, vol. 245, no. April 2022, p. 113663, 2023.

[49] P. J. Schubert, R. Saxena, and J. Kornfeld, "DeepFocus: fast focus and astigmatism correction for electron microscopy," *Nature Communications*, vol. 15, no. 1, pp. 1–10, 2024.

[50] D. Ma, S. E. Zeltmann, C. Zhang, Z. Baraissov, Y.-T. Shao, C. Duncan, J. Maxson, A. Edelen, and D. A. Muller, "Emittance Minimization for Aberration Correction I: Aberration correction of an electron microscope without knowing the aberration coefficients," pp. 1–27, dec 2024.

[51] A. J. Pattison, S. M. Ribet, M. M. Noack, G. Varnavides, K. Park, E. Kirkland, J. Park, C. Ophus, and P. Ercius, "BEACON – Automated Aberration Correction for Scanning Transmission Electron Microscopy using Bayesian Optimization," pp. 1–36, oct 2024.

[52] N. Schnitzer, S. H. Sung, and R. Hovden, "Optimal STEM Convergence Angle Selection Using a Convolutional Neural Network and the Strehl Ratio," *Microscopy and Microanalysis*, vol. 26, no. 5, pp. 921–928, oct 2020.

[53] J. M. Ede, "Deep learning in electron microscopy," *Machine Learning: Science and Technology*, vol. 2, no. 1, p. 011004, mar 2021.

[54] A. Koster, A. Van den Bos, and K. van der Mast, "An autofocus method for a TEM," *Ultramicroscopy*, vol. 21, no. 3, pp. 209–222, jan 1987.

[55] A. Steinecker and W. Mader, "Measurement of lens aberrations by means of image displacements in beam-tilt series," *Ultramicroscopy*, vol. 81, no. 3-4, pp. 149–161, apr 2000.

[56] E. van Horssen, B. Janssen, A. Kumar, D. Antunes, and W. Heemels, "Image-based feedback control for drift compensation in an electron microscope," *IFAC Journal of Systems and Control*, vol. 11, no. 2020, p. 100074, mar 2020.

[57] M. van Meer, T. van Meijel, E. van Halsema, E. Verschueren, G. Witvoet, and T. Oomen, "Compensating hysteresis and mechanical misalignment in piezo-stepper actuators," *Mechatronics*, vol. 111, p. 103394, nov 2025.

[58] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, aug 2013.

[59] N. Cressie and C. K. Wikle, *Statistics for Spatio-Temporal Data*, 1st ed. Hoboken: John Wiley & Sons Inc, 2011.

[60] M. Ghavamzadeh, S. Mannor, J. Pineau, and A. Tamar, "Bayesian Reinforcement Learning: A Survey," *Foundations and Trends® in Machine Learning*, vol. 8, no. 5-6, pp. 359–483, sep 2016.

[61] T. M. Moerland, J. Broekens, A. Plaat, and C. M. Jonker, "Model-based Reinforcement Learning: A Survey," *Foundations and Trends in Machine Learning*, vol. 16, no. 1, pp. 1–118, jan 2023.

[62] J. S. van Hulst, R. A. C. van Zuijlen, N. Javaheri, M. Diephuis, D. J. Antunes, and W. P. M. H. Heemels, "Calibration of Electron Microscopes Through Deep Learning and Bayesian Optimization," *IEEE Access*, vol. 13, pp. 137 291–137 302, 2025.

[63] J. Hillier and E. G. Ramberg, "The Magnetic Electron Microscope Objective: Contour Phenomena and the Attainment of High Resolving Power," *Journal of Applied Physics*, vol. 18, no. 1, pp. 48–71, jan 1947.

[64] J. Orloff, *Handbook of Charged Particle Optics*, 2nd ed., J. Orloff, Ed. CRC Press, 2009.

[65] S. J. Pennycook, "The impact of STEM aberration correction on materials science," *Ultramicroscopy*, vol. 180, pp. 22–33, sep 2017.

[66] P. W. Hawkes and J. C. Spence, Eds., *Science of Microscopy*. New York: Springer, 2007.

[67] S. J. Pennycook, "A Scan Through the History of STEM," in *Scanning Transmission Electron Microscopy*, S. J. Pennycook and P. D. Nellist, Eds. Springer, New York, NY, 2011, pp. 1–90.

[68] N. Schnitzer, S. H. Sung, and R. Hovden, "Introduction to the Ronchigram and its Calculation with Ronchigram.com," *Microscopy Today*, vol. 27, no. 03, pp. 12–15, may 2019.

[69] K. H. Ong, J. C. H. Phang, and J. T. L. Thong, "A robust focusing and astigmatism correction method for the scanning electron microscope," *Scanning*, vol. 19, no. 8, pp. 553–563, 1997.

[70] A. Tejada, S. W. van der Hoeven, A. J. den Dekker, and P. M. J. Van den Hof, "Towards automatic control of scanning transmission electron microscopes," in *2009 IEEE International Conference on Control Applications*. IEEE, jul 2009, pp. 788–793.

[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016.

[72] R. Garnett, *Bayesian Optimization*. Cambridge University Press, jan 2023.

[73] M. E. Rudnaya, W. Van den Broek, R. M. Doornbos, R. M. Mattheij, and J. M. Maubach, "Defocus and twofold astigmatism correction in HAADF-STEM," *Ultramicroscopy*, vol. 111, no. 8, pp. 1043–1054, jul 2011.

[74] J. van Hulst, R. van Zuijlen, D. Antunes, and M. Heemels, "Bayesian Optimization Applied to Calibration of Electron Microscope," in *Benelux Meeting on Systems and Control 2023*, Elspeet, 2023, p. 94.

[75] Y. Lu, X. Zhang, and H. Li, "A simplified focusing and astigmatism correction method for a scanning electron microscope," *AIP Advances*, vol. 8, no. 1, p. 015124, jan 2018.

[76] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, 1st ed. Cambridge, MA: MIT Press, 2006.

[77] X. Yu, J. Wang, Q.-Q. Hong, R. Teku, S.-H. Wang, and Y.-D. Zhang, "Transfer learning for medical images analyses: A survey," *Neurocomputing*, vol. 489, pp. 230–254, jun 2022.

[78] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, T. Dietterich, Ed. MIT Press, 2006.

[79] H. J. Kushner, "A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise," *Journal of Basic Engineering*, vol. 86, no. 1, pp. 97–106, mar 1964.

[80] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, may 2012.

[81] S. Bubeck, R. Munos, and G. Stoltz, "Pure exploration in finitely-armed and continuous-armed bandits," *Theoretical Computer Science*, vol. 412, no. 19, pp. 1832–1852, apr 2011.

[82] P. Hennig and C. J. Schuler, "Entropy Search for Information-Efficient Global Optimization," *Journal of Machine Learning Research*, vol. 13, pp. 1809–1837, dec 2011.

[83] Z. Wang and S. Jegelka, "Max-value Entropy Search for Efficient Bayesian Optimization," *34th International Conference on Machine Learning, ICML 2017*, vol. 7, pp. 5530–5543, jan 2018.

[84] P. Frazier, W. Powell, and S. Dayanik, "The Knowledge-Gradient Policy for Correlated Normal Beliefs," *INFORMS Journal on Computing*, vol. 21, no. 4, pp. 599–613, nov 2009.

[85] J. Villemonteix, E. Vazquez, and E. Walter, "An informational approach to the global optimization of expensive-to-evaluate functions," *Journal of Global Optimization*, vol. 44, no. 4, pp. 509–534, aug 2009.

[86] S. J. Pennycook and P. D. Nellist, Eds., *Scanning Transmission Electron Microscopy*. New York: Springer, 2011.

[87] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization 1998 13:4*, vol. 13, no. 4, pp. 455–492, 1998.

[88] J. S. van Hulst, W. P. M. H. Heemels, and D. J. Antunes, "Bridging the Simulation-to-Reality Gap in Electron Microscope Calibration via VAE-EM Estimation," *Submitted*, mar 2026.

[89] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *Cambridge Explorations in Arts and Sciences*, vol. 2, no. 1, pp. 1–14, dec 2013.

[90] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data Via the EM Algorithm," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 39, no. 1, pp. 1–22, sep 1977.

[91] C. F. J. Wu, "On the Convergence Properties of the EM Algorithm," *The Annals of Statistics*, vol. 11, no. 1, pp. 95–103, mar 1983.

[92] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[93] E. J. Kirkland, *Advanced Computing in Electron Microscopy*, 2nd ed. Boston, MA: Springer US, 2010.

[94] G. K. Nilsen, A. Z. Munthe-Kaas, H. J. Skaug, and M. Brun, "Epistemic uncertainty quantification in deep learning classification by the Delta method," *Neural Networks*, vol. 145, pp. 164–176, jan 2022.

[95] J. Sirignano and K. Spiliopoulos, "Mean field analysis of neural networks: A central limit theorem," *Stochastic Processes and their Applications*, vol. 130, no. 3, pp. 1820–1852, mar 2020.

[96] P. Monchot, L. Coquelin, S. J. Petit, S. Marmin, E. Le Pennec, and N. Fischer, "Input uncertainty propagation through trained neural networks," *Proceedings of Machine Learning Research*, vol. 202, pp. 25 140–25 173, 2023.

[97] R. Bellman and K. Åström, "On structural identifiability," *Mathematical Biosciences*, vol. 7, no. 3-4, pp. 329–339, apr 1970.

[98] E. Walter and L. Pronzato, *Identification of Parametric Models from Experimental Data*, ser. Communications and control engineering. Springer, 1997.

[99] M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur, "Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences," pp. 1–64, jul 2018.

[100] D. Ginsbourger, X. Bay, O. Roustant, and L. Carraro, "Argumentwise invariant kernels for the approximation of invariant functions," *Annales de la Faculté des sciences de Toulouse : Mathématiques*, vol. 21, no. 3, pp. 501–527, oct 2012.

[101] R. M. Neal and G. E. Hinton, "A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants," in *Learning in Graphical Models*. Dordrecht: Springer Netherlands, 1998, pp. 355–368.

[102] J. W. Goodman, *Introduction to Fourier Optics McGraw-Hill Series in Electrical and Computer Engineering*, 3rd ed. Roberts & Company, 2005.

[103] J. S. van Hulst, E. M. Franken, B. J. Janssen, W. M. Heemels, and D. J. Antunes, "Tilt-based Aberration Estimation in Transmission Electron Microscopy," *Mechatronics*, vol. 117, p. 103529, jul 2026.

[104] A. C. Walls, Y.-j. Park, M. A. Tortorici, A. Wall, A. T. McGuire, and D. Veesler, "Structure, Function, and Antigenicity of the SARS-CoV-2 Spike Glycoprotein," *Cell*, vol. 181, no. 2, pp. 281–292.e6, apr 2020.

[105] K. J. Hanszen and L. Trepte, "Die Kontrastübertragung im Elektronenmikroskop bei partiell kohärenter Beleuchtung," pp. 166–182, 1971.

[106] R. M. Glaeser, D. Typke, P. C. Tiemeijer, J. Pulokas, and A. Cheng, "Precise beam-tilt alignment and collimation are required to minimize the phase error associated with coma in high-resolution cryo-EM," *Journal of Structural Biology*, vol. 174, no. 1, pp. 1–10, apr 2011.

[107] F. Pukelsheim, *Optimal Design of Experiments*, R. E. O'Malley, Ed. Society for Industrial and Applied Mathematics, jan 2006.

[108] A. C. Atkinson, A. N. Donev, and R. D. Tobias, *Optimum Experimental Designs, with SAS*. Oxford University PressOxford, may 2007, no. May 2014.

[109] H. Zhang, R. Ayoub, and S. Sundaram, "Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms," *Automatica*, vol. 78, pp. 202–210, apr 2017.

[110] K. You and L. Xie, "Kalman filtering with scheduled measurements," *IEEE Transactions on Signal Processing*, vol. 61, no. 6, pp. 1520–1530, 2013.

[111] C. Li and N. Elia, "Stochastic sensor scheduling via distributed convex optimization," *Automatica*, vol. 58, pp. 173–182, aug 2015.

[112] Z. Sunberg, S. Chakravorty, and R. S. Erwin, "Information Space Receding Horizon Control for Multisensor Tasking Problems," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp. 1325–1336, jun 2016.

[113] J. Barthel, "Ultra-Precise Measurement of Optical Aberrations for Sub-Ångström Transmission Electron Microscopy," PhD Thesis, Technischen Hochschule Aachen, 2003.

[114] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, jun 2010, pp. 2544–2550.

[115] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, mar 1960.

[116] D. Simon, *Optimal State Estimation*. Wiley, may 2006.

[117] C. Rao, J. Rawlings, and D. Mayne, "Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations," *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, feb 2003.

[118] W. H. Kwon, P. S. Kim, and S. H. Han, "A receding horizon unbiased FIR filter for discrete-time state space models," *Automatica*, vol. 38, no. 3, pp. 545–551, mar 2002.

[119] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, jun 2000.

[120] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, oct 2005.

[121] M. Unser, B. L. Trus, and A. C. Steven, "A new resolution criterion based on spectral signal-to-noise ratios," *Ultramicroscopy*, vol. 23, no. 1, pp. 39–51, jan 1987.

[122] J. S. van Hulst, A. M. C. de Peffer, D. Herceg, E. M. Franken, E. Verschueren, W. P. M. H. Heemels, and D. J. Antunes, "Precision Specimen Positioning in Electron Microscopy through Hysteresis Compensation, Iterative Learning, and Vision-Based Sensing," *Submitted*, pp. 1–11, aug 2026.

[123] D. B. Williams and B. C. Carter, *Transmission Electron Microscopy*, 2, Ed. Springer New York, 2009.

[124] J. Li, R. Sedaghati, J. Dargahi, and D. Waechter, "Design and development of a new piezoelectric linear Inchworm actuator," *Mechatronics*, vol. 15, no. 6, pp. 651–681, jul 2005.

[125] A. J. Fleming and K. K. Leang, *Design, modeling and control of nanopositioning systems*, 1st ed. Springer, 2014.

[126] A. Visintin, *Differential Models of Hysteresis*, ser. Applied Mathematical Sciences. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, vol. 111.

[127] P. Krejčí, *Hysteresis, Convexity and Dissipation in Hyperbolic equations*, ser. GAKUTO international series: mathematical sciences and applications. Gakkōtosho, 1996.

[128] N. Strijbosch, K. Tiels, and T. Oomen, "Memory-Element-Based Hysteresis: Identification and Compensation of a Piezoelectric Actuator," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 6, pp. 2863–2870, nov 2023.

[129] L. Aarnoudse, N. Strijbosch, P. Tacx, E. Verschueren, and T. Oomen, "Compensating commutation-angle domain disturbances with application to waveform optimization for piezo stepper actuators," *Mechatronics*, vol. 94, no. February, p. 103016, 2023.

[130] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, no. 3, pp. 96–114, jun 2006.

[131] H.-S. Ahn, Y. Chen, and K. L. Moore, "Iterative Learning Control: Brief Survey and Categorization," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1099–1121, nov 2007.

[132] S. Q. Zheng, E. Palovcak, J.-P. Armache, K. A. Verba, Y. Cheng, and D. A. Agard, "MotionCor2: anisotropic correction of beam-induced motion for improved cryo-electron microscopy," *Nature Methods*, vol. 14, no. 4, pp. 331–332, apr 2017.

[133] W. Ramberg and W. R. Osgood, "Description of stress-strain curves by three parameters," *National Advisory Committee For Aeronautics*, 1943.

[134] D. Fang and J. Liu, "Basic Equations of Piezoelectric Materials," in *Fracture Mechanics of Piezoelectric and Ferroelectric Solids*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 77–95.

[135] R. Pintelon and J. Schoukens, *System Identification: A Frequency Domain Approach*. Wiley, 2012.

[136] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-Domain ILC Approach for Repeating and Varying Tasks: With Application to Semiconductor Bonding Equipment," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 6, pp. 2716–2727, dec 2016.

[137] J. S. van Hulst, J. M. Tomczak, W. P. M. H. Heemels, and D. J. Antunes, "Constructing VAE Latent Spaces with Prescribed Topology," *Submitted*, pp. 1–16, jun 2026.

[138] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *31st International Conference on Machine Learning, ICML 2014*, vol. 4, pp. 3057–3070, 2014.

[139] T. R. Davidson, L. Falorsi, N. De Cao, T. Kipf, and J. M. Tomczak, "Hyperspherical Variational Auto-Encoders," *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, vol. 2, pp. 856–865, apr 2018.

[140] T. R. Davidson, J. M. Tomczak, and E. Gavves, "Increasing Expressivity of a Hyperspherical VAE," in *NeurIPS 2019 Workshop on Bayesian Deep Learning*, vol. 1, no. 1, oct 2019, pp. 1–8.

[141] L. Falorsi, P. de Haan, T. R. Davidson, N. De Cao, M. Weiler, P. Forré, and T. S. Cohen, "Explorations in Homeomorphic Variational Auto-Encoding," in *ICML workshop on Theoretical Foundations and Applications of Deep Generative Models*, jul 2018.

[142] E. Mathieu, C. Le Lan, C. J. Maddison, R. Tomioka, and Y. W. Teh, "Continuous Hierarchical Representations with Poincaré Variational Auto-Encoders," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019.

[143] D. Kalatzis, D. Eklund, G. Arvanitidis, and S. Hauberg, "Variational Autoencoders with Riemannian Brownian Motion Priors," in *ICML 2020*, aug 2020.

[144] L. A. Perez Rey, V. Menkovski, and J. Portegies, "Diffusion Variational Autoencoders," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, vol. 2021-Janua. California: International Joint Conferences on Artificial Intelligence Organization, jul 2020, pp. 2704–2710.

[145] Y. Nagano, S. Yamaguchi, Y. Fujita, and M. Koyama, "A wrapped normal distribution on hyperbolic space for gradient-based learning," *36th International Conference on Machine Learning, ICML 2019*, pp. 8242–8251, 2019.

[146] I. Huh, C. Jeong, J. M. Choe, Y. G. Kim, and D. S. Kim, "Isometric Quotient Variational Auto-Encoders for Structure-Preserving Representation Learning," *Advances in Neural Information Processing Systems*, vol. 36, no. NeurIPS, pp. 1–13, 2023.

[147] A. Kuzina, K. Pratik, F. V. Massoli, and A. Behboodi, "Equivariant Priors for Compressed Sensing with Unknown Orientation," *Proceedings of Machine Learning Research*, vol. 162, pp. 11 753–11 771, 2022.

[148] J. Brehmer and K. Cranmer, "Flows for simultaneous manifold learning and density estimation," *Advances in Neural Information Processing Systems*, nov 2020.

[149] E. Dupont, M. A. Bautista, A. Colburn, A. Sankar, C. Guestrin, J. Susskind, and Q. Shan, "Equivariant Neural Rendering," *37th International Conference on Machine Learning, ICML 2020*, vol. PartF16814, pp. 2741–2750, dec 2020.

[150] M. Moor, M. Horn, B. Rieck, and K. Borgwardt, "Topological Autoencoders," *37th International Conference on Machine Learning, ICML 2020*, vol. PartF16814, pp. 7001–7011, may 2021.

[151] J. M. Tomczak and M. Welling, "VAE with a VampPrior," *Proceedings of Machine Learning Research*, vol. 84, feb 2018.

[152] N. Miao, E. Mathieu, N. Siddharth, Y. W. Teh, and T. Rainforth, "On Incorporating Inductive Biases into VAEs," *ICLR 2022 - 10th International Conference on Learning Representations*, pp. 1–22, feb 2022.

[153] G. Ramachandran, C. Ramakrishnan, and V. Sasisekharan, "Stereochemistry of polypeptide chain configurations," *Journal of Molecular Biology*, vol. 7, no. 1, pp. 95–99, jul 1963.

[154] W. Boomsma, K. V. Mardia, C. C. Taylor, J. Ferkinghoff-Borg, A. Krogh, and T. Hamelryck, "A generative, probabilistic model of local protein structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 26, pp. 8932–8937, jul 2008.

[155] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, may 2006, vol. 9780521862.

[156] G. Carlsson, T. Ishkhanov, V. de Silva, and A. Zomorodian, "On the Local Behavior of Spaces of Natural Images," *International Journal of Computer Vision*, vol. 76, no. 1, pp. 1–12, jan 2008.

[157] I. Gilitschenski, R. Sahoo, W. Schwarting, A. Amini, S. Karaman, and D. Rus, "Deep Orientation Uncertainty Learning based on a Bingham Loss," in *International Conference on Learning Representations*, 2020.

[158] J. M. Lee, *Introduction to Topological Manifolds*, 2nd ed., ser. Graduate Texts in Mathematics. Springer, 2011.

[159] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Wiley, 2006.

[160] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework," in *International Conference on Learning Representations*, 2017.

[161] D. J. Rezende and S. Mohamed, "Variational Inference with Normalizing Flows," *32nd International Conference on Machine Learning, ICML 2015*, vol. 2, pp. 1530–1538, may 2015.

[162] E. Jang, S. Gu, and B. Poole, "Categorical Reparameterization with Gumbel-Softmax," *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, pp. 1–13, aug 2017.

[163] C. J. Maddison, A. Mnih, and Y. W. Teh, "The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables," *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, pp. 1–20, mar 2017.

[164] E. Dupont, "Learning Disentangled Joint Continuous and Discrete Representations," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, no. 2016, pp. 710–720, oct 2018.

[165] B. Sturmfels, *Algorithms in Invariant Theory*, 2nd ed. Springer, 2008.

[166] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition," in *ECCV 2016*, vol. 44, no. 4, 2016, pp. 499–515.

[167] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C. H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, "Deep Unsupervised Clustering with Gaussian Mixture Variational Autoencoders," *arXiv preprint arXiv:1611.02648*, jan 2017.

[168] Y. Burda, R. Grosse, and R. Salakhutdinov, "Importance weighted autoencoders," *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, pp. 1–14, 2016.

[169] J. B. Kruskal, "Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, mar 1964.

[170] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner, "Understanding disentangling in β-VAE," in *NeurIPS 2017 Workshop on Learning Disentangled Representations*, no. Nips, apr 2018.

[171] J. van Hulst, R. van Zuijlen, D. Antunes, and W. M. Heemels, "Estimation of Dynamic Gaussian Processes," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, dec 2023, pp. 3206–3211.

[172] J. S. van Hulst, W. P. M. H. Heemels, and D. J. Antunes, "Estimating Evolving Functions with Dynamic Gaussian Processes," *Submitted*, pp. 1–20, jun 2026.

[173] G. Atluri, A. Karpatne, and V. Kumar, "Spatio-Temporal Data Mining," *ACM Computing Surveys*, vol. 51, no. 4, pp. 1–41, jul 2019.

[174] C. K. Wikle, "Modern perspectives on statistics for spatio-temporal data," *WIREs Computational Statistics*, vol. 7, no. 1, pp. 86–98, jan 2015.

[175] F. Lutscher, *Integrodifference equations in spatial ecology*. Springer, 2018.

[176] C. K. Wikle and N. Cressie, "A Dimension-Reduced Approach to Space-Time Kalman Filtering," *Biometrika*, vol. 86, no. 4, pp. 815–829, 1999.

[177] D. Rozier, F. Birol, E. Cosme, P. Brasseur, J. M. Brankart, and J. V. Verron, "A reduced-order Kalman filter for data assimilation in physical oceanography," *SIAM Review*, vol. 49, no. 3, pp. 449–465, 2007.

[178] G. Evensen, "The Ensemble Kalman Filter: theoretical formulation and practical implementation," *Ocean Dynamics*, vol. 53, no. 4, pp. 343–367, nov 2003.

[179] K. J. H. Law, A. M. Stuart, and K. C. Zygalakis, *Data Assimilation: A Mathematical Introduction*, jun 2015.

[180] A. M. Stuart, "Inverse problems: A Bayesian perspective," *Acta Numerica*, vol. 19, pp. 451–559, may 2010.

[181] S. Särkkä, A. Solin, and J. Hartikainen, "Spatiotemporal Learning via Infinite-Dimensional Bayesian Filtering and Smoothing: A Look at Gaussian Process Regression Through Kalman Filtering," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 51–61, jul 2013.

[182] M. Todescato, A. Carron, R. Carli, G. Pillonetto, and L. Schenato, "Efficient spatio-temporal Gaussian regression via Kalman filtering," *Automatica*, vol. 118, p. 109032, aug 2020.

[183] A. Carron, M. Todescato, R. Carli, L. Schenato, and G. Pillonetto, "Machine learning meets Kalman Filtering," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, vol. 1, no. Cdc. IEEE, dec 2016, pp. 4594–4599.

[184] A. Solin and S. Särkkä, "Hilbert space methods for reduced-rank Gaussian process regression," *Statistics and Computing*, vol. 30, no. 2, pp. 419–446, mar 2020.

[185] N. Cressie and G. Johannesson, "Fixed rank kriging for very large spatial data sets," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 1, pp. 209–226, jan 2008.

[186] N. Cressie, T. Shi, and E. L. Kang, "Fixed Rank Filtering for Spatio-Temporal Data," *Journal of Computational and Graphical Statistics*, vol. 19, no. 3, pp. 724–745, jan 2010.

[187] K. V. Mardia, C. Goodall, E. J. Redfern, and F. J. Alonso, "The Kriged Kalman filter," *Test*, vol. 7, no. 2, pp. 217–282, dec 1998.

[188] R. Curtain, "A Survey of Infinite-Dimensional Filtering," *SIAM Review*, vol. 17, no. 3, pp. 395–411, jul 1975.

[189] A. Pazy, *Semigroups of Linear Operators and Applications to Partial Differential Equations*. Springer, 1996, vol. 115, no. 399.

[190] L. Evans, *Partial Differential Equations*, ser. Graduate studies in mathematics. American Mathematical Society, 2010.

[191] B. F. Farrell and P. J. Ioannou, "State estimation using a reduced-order Kalman filter," *Journal of the Atmospheric Sciences*, vol. 58, no. 23, pp. 3666–3680, 2001.

[192] B. Anderson and J. Moore, *Optimal Filtering*, 2nd ed., ser. Dover Books on Engineering. Dover Publications, 2005.

[193] T. Janjić, N. Bormann, M. Bocquet, J. A. Carton, S. E. Cohn, S. L. Dance, S. N. Losa, N. K. Nichols, R. Potthast, J. A. Waller, and P. Weston, "On the representation error in data assimilation," *Quarterly Journal of the Royal Meteorological Society*, vol. 144, no. 713, pp. 1257–1278, apr 2018.

[194] M. Reed and B. Simon, *Methods of Modern Mathematical Physics*. Elsevier, 1972.

[195] R. Kress, *Linear Integral Equations*, 2nd ed., ser. Applied Mathematical Sciences. New York, NY: Springer New York, 1999, vol. 82.

[196] G. Da Prato and J. Zabczyk, *Stochastic Equations in Infinite Dimensions*, 2nd ed., ser. Encyclopedia of Mathematics and its Applications. Cambridge University Press, apr 2014.

[197] J. Hensman, N. Durrande, and A. Solin, "Variational Fourier Features for Gaussian Processes," *Journal of Machine Learning Research*, vol. 18, pp. 1–52, 2018.

[198] D. Bernstein and D. Hyland, "The optimal projection equations for reduced-order state estimation," *IEEE Transactions on Automatic Control*, vol. 30, no. 6, pp. 583–585, jun 1985.

[199] A. Germani, L. Jetto, and M. Piccioni, "Galerkin Approximation for Optimal Linear Filtering of Infinite-Dimensional Linear Systems," *SIAM Journal on Control and Optimization*, vol. 26, no. 6, pp. 1287–1305, nov 1988.

[200] J. A. Burns and J. Cheung, "Optimal convergence rates for Galerkin approximation of operator Riccati equations," *Numerical Methods for Partial Differential Equations*, vol. 38, no. 6, pp. 2045–2083, nov 2022.

[201] J. van Hulst, W. Heemels, and D. Antunes, "Data-Efficient Quadratic Q-Learning Using LMIs," in *2024 IEEE 63rd Conference on Decision and Control (CDC)*. Milan: IEEE, dec 2024, pp. 1161–1166.

[202] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press, 2018.

[203] L. Buşoniu, D. Ernst, B. De Schutter, and R. Babuška, "Approximate reinforcement learning: An overview," *IEEE SSCI 2011: Symposium Series on Computational Intelligence - ADPRL 2011: 2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning*, pp. 1–8, 2011.

[204] L. Buşoniu, T. de Bruin, D. Tolić, J. Kober, and I. Palunko, "Reinforcement learning for control: Performance, stability, and deep approximators," *Annual Reviews in Control*, vol. 46, pp. 8–28, 2018.

[205] X. Xu, L. Zuo, and Z. Huang, "Reinforcement learning algorithms with function approximation: Recent advances and applications," *Information Sciences*, vol. 261, pp. 1–31, mar 2014.

[206] S. J. Bradtke and A. G. Barto, "Linear Least-Squares algorithms for temporal difference learning," *Machine Learning*, vol. 22, no. 1-3, pp. 33–57, 1996.

[207] M. G. Lagoudakis and R. Parr, "Least-squares policy iteration," *Journal of Machine Learning Research*, vol. 4, no. 6, pp. 1107–1149, 2003.

[208] M. G. Lagoudakis, "Least-Squares Reinforcement Learning Methods," in *Encyclopedia of Machine Learning and Data Mining*. Boston, MA: Springer US, 2014, pp. 1–9.

[209] D. Ernst, P. Geurts, and L. Wehenkel, "Tree-based batch mode reinforcement learning," *Journal of Machine Learning Research*, vol. 6, pp. 503–556, 2005.

[210] M. Riedmiller, "Neural Fitted Q Iteration – First Experiences with a Data Efficient Neural Reinforcement Learning Method," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2005, vol. 3720 LNAI, pp. 317–328.

[211] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, feb 2015.

[212] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint arXiv:1707.06347*, jul 2017.

[213] A. Abdolmaleki, J. T. Springenberg, Y. Tassa, R. Munos, N. Heess, and M. Riedmiller, "Maximum a Posteriori Policy Optimisation," *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, jun 2018.

[214] D. Bertsekas and J. N. Tsitsiklis, *Neuro-dynamic programming.* Belmont, Massachusetts: Athena Scientific, 1996.

[215] L. Baird, "Residual Algorithms: Reinforcement Learning with Function Approximation," in *Machine Learning Proceedings 1995.* Elsevier, 1995, pp. 30–37.

[216] C. Sanathanan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Transactions on Automatic Control*, vol. 8, no. 1, pp. 56–58, jan 1963.

[217] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, jan 2010.

[218] L. El Ghaoui, F. Oustry, and M. AitRami, "A cone complementarity linearization algorithm for static output-feedback and related problems," *IEEE Transactions on Automatic Control*, vol. 42, no. 8, pp. 1171–1176, 1997.

[219] R. A. Freeman and P. V. Kokotovic, "Inverse Optimality in Robust Stabilization," *SIAM Journal on Control and Optimization*, vol. 34, no. 4, pp. 1365–1391, jul 1996.

[220] J. S. van Hulst, W. P. M. H. Heemels, and D. J. Antunes, "Smart Exploration in Reinforcement Learning Using Bounded Uncertainty Models," in *2025 IEEE 64th Conference on Decision and Control (CDC).* Rio de Janeiro: IEEE, dec 2025, pp. 5132–5138.

[221] M. Kearns and S. Singh, "Near-Optimal Reinforcement Learning in Polynomial Time," *Machine Learning*, vol. 49, pp. 209–232, 2002.

[222] R. S. Sutton, "Dyna, an integrated architecture for learning, planning, and reacting," *ACM SIGART Bulletin*, vol. 2, no. 4, pp. 160–163, jul 1991.

[223] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, Bellevue, WA, 2011, pp. 465–472.

[224] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan, "Is Q-learning Provably Efficient?" *Advances in Neural Information Processing Systems*, vol. 2018-Decem, pp. 4863–4873, jul 2018.

[225] T. Jaksch, R. Ortner, and P. Auer, "Near-optimal regret bounds for reinforcement learning," *Journal of Machine Learning Research*, vol. 11, pp. 1563–1600, 2010.

[226] I. Osband, D. Russo, and B. Van Roy, "(More) Efficient Reinforcement Learning via Posterior Sampling," *Advances in Neural Information Processing Systems*, pp. 1–10, dec 2013.

[227] M. Ghavamzadeh, S. Mannor, J. Pineau, and A. Tamar, "Bayesian Reinforcement Learning: A Survey," *Foundations and Trends® in Machine Learning*, vol. 8, no. 5-6, pp. 359–483, nov 2015.

[228] R. Givan, S. Leach, and T. Dean, "Bounded-parameter Markov decision processes," *Artificial Intelligence*, vol. 122, no. 1-2, pp. 71–109, sep 2000.

[229] S. Haddad and B. Monmege, "Interval iteration algorithm for MDPs and IMDPs," *Theoretical Computer Science*, vol. 735, pp. 111–131, jul 2018.

[230] M. Petrik and R. H. Russell, "Beyond Confidence Regions: Tight Bayesian Ambiguity Sets for Robust MDPs," *Advances in Neural Information Processing Systems*, vol. 32, pp. 1–23, feb 2019.

[231] D. Bertsekas and S. E. Shreve, *Stochastic Optimal Control: The Discrete-Time Case*, ser. Athena scientific optimization and computation series. Athena Scientific, 1996.

[232] D. P. Bertsekas, *Abstract Dynamic Programming*, 3rd ed. Athena Scientific, 2022.

[233] W. Hoeffding, "Probability Inequalities for Sums of Bounded Random Variables," *Journal of the American Statistical Association*, vol. 58, no. 301, p. 13, mar 1963.

[234] A. L. Strehl, H. Li, and M. L. Littman, "Reinforcement learning in finite MDPs: PAC analysis," *Journal of Machine Learning Research*, vol. 10, pp. 2413–2444, 2009.

[235] J. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Transactions on Automatic Control*, vol. 42, no. 5, pp. 674–690, may 1997.

[236] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "OpenAI Gym," pp. 1–4, jun 2016.

[237] J. van Hulst, M. Poot, D. Kostić, K. W. Yan, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities: A Learning-based Approach," in *IFAC-PapersOnLine*, vol. 55, no. 37. Elsevier, jan 2022, pp. 235–240.

[238] M. Poot, J. van Hulst, K. W. Yan, D. Kostić, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities: With Application to a Wirebonder," in *IFAC-PapersOnLine*, Yokohama, 2023, pp. 1895–1900.

[239] R. van Zuijlen, J. van Hulst, W. Heemels, and D. Antunes, "Bayesian Estimation for Linear Systems with Nonlinear Output and General Noise Distribution Using Fourier Basis Functions," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, dec 2023, pp. 2166–2171.

[240] J. van Hulst, W. P. M. H. M. Heemels, and D. J. Antunes, "Faster Quadratic Q-Learning by Exploiting Linear Matrix Inequalities," in *Benelux Meeting on Systems and Control 2024*, Blankenberge, 2024, p. 129.

[241] T. Ickenroth, M. van Haren, J. Kon, M. van Meer, J. van Hulst, and T. Oomen, "Automatic Basis Function Selection in Iterative Learning Control : A Sparsity-Promoting Approach Applied to an Industrial Printer," in *2025 American Control Conference (ACC)*, 2025, pp. 2931–2936.

[242] J. van Hulst, M. Poot, D. Kostić, K. W. Yan, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities : A Learning-based Approach," in *Benelux Meeting on Systems and Control 2022*, Brussels, 2022, p. 130.

[243] J. S. van Hulst, W. M. Heemels, and D. J. Antunes, "Data-driven Aberration Estimation in Electron Microscopes," in *Benelux Meeting on Systems and Control 2025*, Egmond aan Zee, 2025, p. 122.

[244] J. S. van Hulst, E. M. Franken, B. J. Janssen, W. P. M. H. Heemels, and D. J. Antunes, "Fast Tilt-Based Electron Microscope Alignment Using A-Optimal Sensor Scheduling," in *Benelux Meeting on Systems and Control 2026*, Lommel, 2026, p. 90.

List of Publications

Peer-Reviewed Journal Articles

- J.S. van Hulst, R.A.C. van Zuijlen, N. Javaheri, M. Diephuis, D.J. Antunes, and W.P.M.H. Heemels, "Calibration of Electron Microscopes Through Deep Learning and Bayesian Optimization," *IEEE Access*, vol. 13, pp. 137 291–137 302, 2025. (Chapter 2)

- J.S. van Hulst, E.M. Franken, B.J. Janssen, W.P.M.H. Heemels, and D.J. Antunes, "Tilt-Based Aberration Estimation in Transmission Electron Microscopy," *IFAC Mechatronics*, vol. 117, p. 103529, 2026. (Chapter 4)

- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Bridging the Simulation-to-Reality Gap in Electron Microscope Calibration via VAE-EM Estimation," under review, arXiv:2603.16549, 2026. (Chapter 3)

- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Estimating Evolving Functions with Dynamic Gaussian Processes," under review, arXiv:2606.06705, 2026. (Chapter 7)

- J.S. van Hulst, A.M.C. de Peffer, D. Herceg, E.M. Franken, E. Verschueren, W.P.M.H. Heemels, and D.J. Antunes, "Precision Specimen Positioning in Electron Microscopy through Hysteresis Compensation, Iterative Learning, and Vision-Based Sensing," under review, arXiv:2608.03669, 2026. (Chapter 5)

Peer-Reviewed Articles in Conference Proceedings

- J.S. van Hulst, M. Poot, D. Kostić, K.W. Yan, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities: A Learning-based Approach," in *IFAC-PapersOnLine* (Modeling, Estimation and Control Conference (MECC) 2022), vol. 55, no. 37, pp. 235–240, 2022.

- M. Poot, J.S. van Hulst, K.W. Yan, D. Kostić, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities: With Application to a Wirebonder," in *IFAC-PapersOnLine* (22nd IFAC World Congress), vol. 56, no. 2, pp. 1895–1900, 2023.

- J.S. van Hulst, R.A.C. van Zuijlen, D.J. Antunes, and W.P.M.H. Heemels, "Estimation of Dynamic Gaussian Processes," in *Proc. 62nd IEEE Conference on Decision and Control (CDC)*, Singapore, 2023, pp. 3206–3211. (Chapter 7)

- R.A.C. van Zuijlen, J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Bayesian Estimation for Linear Systems with Nonlinear Output and General Noise Distribution Using Fourier Basis Functions," in *Proc. 62nd IEEE Conference on Decision and Control (CDC)*, Singapore, 2023, pp. 2166–2171.

- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Data-Efficient Quadratic Q-Learning Using LMIs," in *Proc. 63rd IEEE Conference on Decision and Control (CDC)*, Milan, 2024, pp. 1161–1166. (Chapter 8)

- T. Ickenroth, M. van Haren, J. Kon, M. van Meer, J.S. van Hulst, and T. Oomen, "Automatic Basis Function Selection in Iterative Learning Control: A Sparsity-Promoting Approach Applied to an Industrial Printer," in *Proc. 2025 American Control Conference (ACC)*, Denver, 2025, pp. 2931–2936.

- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Smart Exploration in Reinforcement Learning Using Bounded Uncertainty Models," in *Proc. 64th IEEE Conference on Decision and Control (CDC)*, Rio de Janeiro, 2025, pp. 5132–5138. (Chapter 9)

- J.S. van Hulst, J.M. Tomczak, W.P.M.H. Heemels, and D.J. Antunes, "Constructing VAE Latent Spaces with Prescribed Topology," under review, arXiv:2606.07058, 2026. (Chapter 6)

Abstracts in Conference Proceedings

- J.S. van Hulst, M. Poot, D. Kostić, K.W. Yan, J. Portegies, and T. Oomen, "Feedforward Control in the Presence of Input Nonlinearities: A Learning-based Approach," in *Benelux Meeting on Systems and Control*, Brussels, 2022, p. 130.

- J.S. van Hulst, R.A.C. van Zuijlen, D.J. Antunes, and W.P.M.H. Heemels, "Bayesian Optimization Applied to Calibration of Electron Microscope," in *Benelux Meeting on Systems and Control*, Elspeet, 2023, p. 94.

- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Faster Quadratic Q-Learning by Exploiting Linear Matrix Inequalities," in *Benelux Meeting on Systems and Control*, Blankenberge, 2024, p. 129.
- J.S. van Hulst, W.P.M.H. Heemels, and D.J. Antunes, "Data-driven Aberration Estimation in Electron Microscopes," in *Benelux Meeting on Systems and Control*, Egmond aan Zee, 2025, p. 122.
- J.S. van Hulst, E.M. Franken, B.J. Janssen, W.P.M.H. Heemels, and D.J. Antunes, "Fast Tilt-Based Electron Microscope Alignment Using A-Optimal Sensor Scheduling," in *Benelux Meeting on Systems and Control*, Lommel, 2026, p. 90.

Dankwoord

Jeetje, dat was het dan! De afgelopen jaren viel mijn werk soms bijna samen met wie ik ben. Ik ben ontzettend trots op deze thesis, ook al zullen de meesten van jullie dit boekje nooit van kaft tot kaft lezen (sorry voor degenen die dat wel doen, ik trakteer jullie wel een keer op een ijsje). Dit boekje is een tijdscapsule van deze periode van mijn leven, en ik ben blij dat het op die manier altijd zal blijven bestaan. Erin zit mijn vreugde, mijn ontwikkeling, mijn verdriet, maar ook de vele reisjes, feestjes, en ontspannen momenten met familie en vrienden. Zonder de belangrijke mensen uit mijn leven was dit nooit gelukt. Ik ben dankbaar voor alle hulp, steun, en inspiratie die ik van jullie heb gekregen.

First of all, Duarte. We've had over four years of weekly meetings, countless emails, and many hours of discussion. I appreciate how absolutely passionate about your work you are. You once gave me feedback on a paper at 02:21. You would get up early in the morning to commute from Brussels to Eindhoven, and however tired you were, you would suddenly become completely energized when we talked about football and especially Portugal. At the same time, I truly appreciate the respect and care you show for your students. I feel like I could always be open and honest with you, and that you would never judge me for it.

Maurice, bedankt voor je begeleiding en alle input op mijn onderzoek. Bij System Theory hebben we ondertussen heel wat voorbij zien komen: van leuke vragen van studenten en gemotiveerde TAs, tot studenten die aan ons vragen waarom ChatGPT een ander antwoord geeft dan het boek. Ik ben ontzettend onder de indruk van je diepgaande analytische kennis, die je naadloos combineert met een mensgerichte aanpak wat betreft je studenten. Maar desondanks denk ik dat je al die *best teacher awards* niet zou hebben gewonnen zonder jouw gevoel voor humor.

Erik, bedankt voor alle middelen die je voor ons hebt vrijgemaakt om onze ideeën te realiseren. Onze niet-werk-gerelateerde discussies tijdens de lunchpauzes op de maandagen kon ik altijd goed waarderen: engineer-denken toepassen op andere facetten van het leven. Ik vond het altijd leuk om je te horen lachen om de eigenaardigheden van taal en om je soms verwoestende kritiek op de politiek. Je bent het soort nerd dat ik zelf ooit hoop te worden: deskundig, nieuwsgierig, en

schaamteloos enthousiast.

I would also like to thank all the other members of my PhD committee, dr. L. Özkan, prof. dr. K.J. Batenburg, Prof. S. Särkkä, and Prof. Dr. S. Trimpe. Thank you for your time and effort, and for the sharp questions that will undoubtedly be asked during my defense.

Moreover, I would like to thank all members of the ASIMOV and Learning in Motion consortia. In particular, I am grateful for the fruitful collaboration with Thermo Fisher Scientific, enabled especially by Bart Janssen, Edwin Verschueren, Domagoj Herceg, Fouzia Khan, Maurits Diephuis, and Narges Javaheri. Your domain expertise and the many hours spent on discussions and experiments have been invaluable for my research. I would like to thank Jakub Tomczak for the intercontinental collaboration and the input on the VAE chapter. I would also like to thank the MSc students who contributed to my research: Joyce, Tim, Ataur, and Stan.

Naast de inhoudelijke samenwerkingen, zijn er binnen de TU/e ook veel mensen geweest die mijn PhD-tijd op andere manieren hebben verrijkt. Ik ben dit avontuur begonnen in de krochten van Gemini-Zuid, samen met Roeland, Lars, Bas, en Roy. Ik wil jullie bedanken voor de dagelijkse lunchsprints, de tijdige koffiepauzes, de eindeloze schaakpotjes, het dagelijkse Worldle-moment, onze AIVD-kerstpuzzelpoging, en het begin van FC Overshoot. Na bijna drie jaar in deze kelder heb ik eindelijk weer wat daglicht gezien, en heb ik het einde van mijn PhD meegemaakt in Pendulum 3.10.

In het bijzonder wil ik een aantal mensen even uitlichten. Als eerste Maurice, de dromer. Jouw aanstekelijke enthousiasme en je talent om altijd de gekste avonturen te beleven hebben volgens mij niet alleen bij mij een rol gespeeld in de keuze om een PhD te doen. Max, je puppy-achtige enthousiasme, je lijkt wel altijd vrolijk te zijn. Ik waardeer onze discussies waarbij we *niets* voor waar aannemen, en alles kritisch aanpakken. Mathyn, je directe, concrete, soms harde, voeten-op-de-grond blik op het leven. Bedankt voor al het slappe gezever, maar ook voor de diepe gesprekken op de fiets naar huis na het fitnessen. Door deze gym rattata's heb ik tijdens mijn promotie een nieuwe hobby gevonden, waardoor ik veerkrachtiger en zekerder ben geworden. Over fitnessfanaten gesproken, Menno! Ik bedank je voor je controversiële gesprekstarters, die verhullen dat je eigenlijk ontzettend betrokken en zorgzaam bent. Verder wil ik nog graag bedanken: Sven voor zijn punctualiteit, (Hairy) Max voor zijn goede zorgen over Schnautzi, Lennard voor zijn vurige voetbaltalent, Tjeerd voor zijn kickroll, Tren voor zijn coole koelbox én wijnkoelkast, Maarten voor zijn kennis over feestjes in Milaan, Timo voor zijn aanstekelijke dansmoves, En Eline voor het delen van de kantoorsleutel, al is het me nog steeds een raadsel hoe je erbij kon. En natuurlijk Nancy, Lindsey, en Roos, door wie elke e-mail leuker, elke borrel gezelliger, en elke sportdag fanatieker werd.

Daarnaast wil ik ook graag mijn vrienden en familie bedanken, die me vooral

de afgelopen jaren ongelofelijk veel plezier hebben gegeven. Bart, Max, Gijs, Kim, en Steven voor de vele reizen door Nederland, de spelletjesavonden, de zomerse kerstdiners. De Bennies extended universe: Konrad en Anouk, Jens en Sterre (Stens), Remco en Fay, Rob en Luc, Joep, Thom en Marloes (en Ben), Pim en Jessie (en Nova), Bas en Feline, Reshabh, Eva en Jurre, Kevin en Lot. De festivals, de verjaardagen, de vakanties, de vriendenweekenden, samen sporten, samen de stad in. Ik kijk er elke keer weer naar uit om jullie te zien.

Ik wil graag mijn paranimfen Joep en Max bedanken. Ik heb in deze periode van mijn leven veel van mezelf met jullie gedeeld, en daar was altijd ruimte voor. Ik vind het heel vet dat we deze dag samen kunnen vieren.

Sjoerd, ik wil je bedanken voor je vriendschap, voor je betrokkenheid, en voor je bulderende lach. Bente, wat is het fijn om met jou te praten als je iets dwars zit. Jij hebt dan misschien wel gymnasium gedaan, maar ik heb nu een PhD, dus... Crit en Just, bedankt voor de vele potjes voetbal waardoor ik mijn hoofd leeg kon maken. Pap en mam, ik wil jullie bedanken voor jullie vertrouwen. Jullie hebben me geleerd om goed na te denken over dingen: nou dat heb ik gedaan ;) Door jullie heb ik het gevoel dat ik mag zijn wie ik ben, en dat ik mag doen wat ik leuk vind.

En dan natuurlijk Hanne. Zonder jou had ik dit niet gekund. Echt waar. Wat ben jij betrokken geweest bij dit avontuur. Ik heb het gevoel dat je altijd weet wat er in mij omgaat. En wat waren we ruim vier jaar geleden nog op een andere plek in ons leven. Nog vóór onze verhuizing naar Nijmegen. Nog vóór onze verhuizing naar Eindhoven. Nog vóór onze eerste baan, vóór onze bruiloft, vóór onze eerste Lowlands, vóór mijn eerste paper. Toch waren we toen, net als nu, twee mensen die hun best doen voor elkaar. Na mijn verdediging blijft er eigenlijk nog maar één echte 'vóór' over, en ik kan niet wachten om ook dat avontuur met jou aan te gaan.

About the Author

Jilles van Hulst was born on March 3, 1998, in Veldhoven, the Netherlands. After finishing his secondary education in 2015 at the Sondervick College in Veldhoven, he studied Mechanical Engineering at the Eindhoven University of Technology, where he received the Bachelor of Science degree and Master of Science degree in 2019 and 2022, respectively. His master thesis focused on rational and nonlinear data-driven feedforward control for wire bonders, under the supervision of Tom Oomen and in collaboration with ASM Pacific Technology.

In 2022, Jilles started his PhD research at the Control Systems Technology Section of the Department of Mechanical Engineering at the Eindhoven University of Technology, under the supervision of Maurice Heemels and Duarte Antunes. The research was carried out in collaboration with Thermo Fisher Scientific, with Erik Franken as co-supervisor. The main results of this research are included in this thesis and focus on learning-based control and estimation for electron microscopy, with an emphasis on data efficiency. The research was part of the ITEA4 project entitled ASIMOV and the research project Learning in Motion, both of which are public-private partnerships with industry partners including Thermo Fisher Scientific.