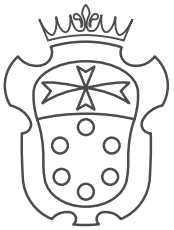


SCUOLA
NORMALE
SUPERIORE

Classe di Scienze

Corso di perfezionamento in Metodi computazionali e modelli matematici per le scienze e la finanza

XXXVI ciclo

Reinforcement learning and numerical methods applied to some quantum information problems

Settore Scientifico Disciplinare **FIS/02**

Candidato
dr. Fabio Zoratti

Relatore
Prof. Vittorio Giovannetti

Anno accademico 2023–2024

Abstract

This thesis explores diverse aspects of quantum information processing, addressing challenges in quantum communication, error correction, and metrology. The study begins with a study of a variational quantum algorithm designed to enhance quantum error correction techniques. Employing an unconventional distance to overcome the issue of barren plateaus, the algorithm showcases improved scalability and performance. This novel distance is the Quantum Wasserstein distance of order 1, which lacks a very common property, the unitary invariance. This property will be crucial to tame the barren plateaus phenomenon.

Moving forward, this work includes an extensive analysis of the Dolinar receiver. We proposed a generalization to the Quantum State Discrimination problem, in which we relaxed one of its hypotheses, proposing then a solution to this generalized counterpart. The investigation provides a theoretical foundation for improving the discrimination capabilities of quantum states, with potential applications in quantum communication protocols. In particular, this new set of hypotheses can be extremely useful for quantum communication over very noisy optical quantum channels, and can also help in the problem of phase locking.

Furthermore, the thesis introduces a general methodology grounded in reinforcement learning principles to tackle experimental design challenges in quantum metrology. This approach is manifested in the form of a Python library built on TensorFlow, facilitating the optimization of quantum measurements for enhanced precision and accuracy. The proposed framework offers a versatile tool for researchers and experimentalists working in quantum metrology, streamlining the process of designing experiments tailored to specific quantum systems.

In summary, this thesis addresses some challenges in quantum information using numerical methods and provides innovative solutions that contribute to the ongoing development of the field.

Contents

Chapter 1

Introduction

Quantum mechanics, the cornerstone of modern physics, describes the behavior of particles at the atomic and subatomic levels, whereas classical mechanics fails to provide accurate predictions. The mathematical formulation of quantum mechanics relies heavily on differential equations, linear algebra, and complex functions, which often lead to complex problems that are analytically intractable. This is where numerical methods come into play, offering powerful tools to approximate solutions to these otherwise unsolvable equations and providing a deeper understanding of quantum phenomena.

Numerical methods are algorithms used for solving mathematical problems that cannot be addressed by exact analytical techniques. In the context of quantum mechanics, these methods enable physicists to tackle problems ranging from the behavior of single particles to the interactions in complex systems. The inherent complexity of quantum systems, characterized by the probabilistic nature of particle states, superposition principles, and the unavoidable presence of noise, makes numerical solutions indispensable for theoretical research and practical applications.

However, the scope of this text is not to provide a comprehensive characterization of all the applications of numerical methods to quantum mechanics. Instead, we will focus on specific examples within emerging and impactful fields: quantum metrology, quantum sensing, quantum control, and quantum error correction. These areas are at the forefront of research and technological development, where numerical methods offer vital tools for progress and innovation. The work in this thesis is divided as follows:

- Firstly, in Chapter 2, we provide a concise overview of the foundational concepts and technical prerequisites essential for comprehending the subsequent discussions in this work. This chapter primarily focuses on the background of Quantum Information, which is considered the

central theme and pivotal area of study for the topics that follow. It includes key principles and theoretical frameworks that form the basis of understanding quantum systems and their information processing capabilities, thus setting the stage for the more specialized applications and methodologies discussed in the later chapters.

- In Chapter 3, we deal with the problem of Barren Plateaus in the context of Variational Quantum Algorithms (VQA, applied to Quantum Error Correction (QEC). As explained in detail later, the natural choices for cost functions in VQA have a very flat landscape everywhere except for some small regions near local and global minima. This situation is undesirable for the convergence of such algorithms. Gradient-based methods rely heavily on a good computation of the first derivatives of the cost function. With a flat landscape, the gradient becomes almost zero in most of the domain, leaving the minimizer with no useful direction where to go. We decided to explore a nonconventional cost function, that lacks a key property of all-natural choices, the invariance under unitary transformation, trying to tackle this problem. This function is the Quantum Wasserstein distance of order 1, recently proposed, that encodes the property that "two states are close if they differ only by a few qubits". This feature may modify heavily the flat landscape shape, reducing the barren plateau phenomenon.

  The comparison has not been performed on a theoretical level, rather than with a practical example, in the context of QEC. Most of these QEC procedures are tailored to a very specific kind of noise, so we tried to devise an agnostic procedure, that can be trained for a specific quantum circuit. This chapter is mainly based on [ZDPK+23].

- In Chapter 4 we consider a generalization of a well-known problem in quantum information, namely quantum state discrimination. The original problem has already been theoretically solved by Helstrom a long time ago, but the solution has an almost impossible practical implementation for several examples of interest. One of the few cases where an experimental implementation has been performed is the Dolinar receiver, where the task is to distinguish between two different coherent states. We generalized the original Helstrom problem, removing one of its hypotheses, about the classical knowledge of the states to discriminate and tried to solve it in the same context as the Dolinar receiver. This extension of the problem may be useful for communication over very noisy channels and to tackle the problem of phase-locking in this kind of laser communication setting. This chapter is mainly based on [ZPFG21].

- In Chapter 5 we tried to close the gap to the theoretical bound of the problem tackled in the former chapter, and generalize the approach to a much broader class of problems.

  A practical implementation of the Dolinar receiver requires performing some fast-paced feedback on a quantum system of interest, where the knowledge acquired on it in the first n measurements is used entirely to decide how to control and measure the system for the $n+1$ measurement. This is a very common situation in quantum metrology and quantum sensing, and for this reason, we decided to study a general method to tackle this kind of problems. This new implementation mixes Bayesian probability with reinforcement learning to create an agent that can adaptively acquire knowledge on a quantum system of interest via the classical outcomes of measurements, and decide how to control the system to maximize the knowledge acquired from the measurements, which can be also tuned accordingly.

  The theoretical framework has been implemented as a Python library based on TensorFlow and has been utilized to study various examples of interest. While most of these examples are not included in this thesis, they can be found in the two papers this chapter is based on, namely [BZMG, BZG24b]. In this thesis, we concentrate on the results achieved with this library for the Dolinar receiver, where the gap with the theoretical performance has been nearly closed.

Finally, we conclude by discussing the results obtained during this PhD and provide an overview of potential directions for future research.

Chapter 2

Technical background

2.1 Quantum mechanics

2.1.1 Measurements

To extract classical information from a quantum state we have to perform some measurements. In quantum information theory, the formalism of von Neumann projective measurements is usually replaced by the more general formalism of the POVMs: positive operator-valued measurement. A POVM is a set of positive operators labelled by an index. These operators have to fulfil a normalization constraint

$$\sum_i \hat{E}_i = \hat{\mathbb{1}}.$$

At each POVM operator corresponds a measurement outcome o_i. The probability of obtaining the outcome o_i on a given quantum state $\hat{\rho}$ is given by the generalized Born rule

$$p(o_i) = \mathrm{tr}[\hat{E}_i \hat{\rho}].$$

The state after the measurement will non-linearly collapse on the state

$$\hat{\rho} \to \frac{\hat{M}_k \hat{\rho} \hat{M}_k^\dagger}{\mathrm{tr}[\hat{M}_k \hat{\rho} \hat{M}_k^\dagger]} \qquad \hat{M}_k = \hat{U}_k \sqrt{\hat{E}_k},$$

where the matrix $\hat{U}_k$ is a generic unitary matrix. To have a feel of the reason why this matrix is necessary, we can consider the following realistic

example. We have at our disposal an optical fibre. Within this fibre, there can be a photon $|1\rangle$ or no photons $|0\rangle$. We use a photon-detector to detect the presence of a photon in this fibre cable. After the measurement, either the photon-counter clicked or not, the fibre will be in the vacuum state $|0\rangle$. If we only projected with the operator $\hat{\rho} \to \sqrt{\hat{E}_k}\hat{\rho}\sqrt{\hat{E}_k}$ we would have obtained an inconsistency.

2.1.2 Quantum state tomography

Given a single copy of a quantum state $\hat{\rho}$ and no extra information, there does not exist a POVM that allows us to reconstruct the structure of this state $\hat{\rho}$ with certainty, and this is due to two reasons

- After a measurement, the state will be modified accordingly, usually destroying a lot of extra information.
- We have pieces of information only relative to the POVM we chose.

However, if we have a sufficiently large number of copies of this state, meaning that we have at our disposal the state $\hat{\rho}^{\otimes n}$, then there exist some methods to perform this reconstruction, performing measurements with respect to several POVMs. A procedure of this kind is called *quantum state tomography*.

2.2 Quantum state discrimination

Since its early studies, the discrimination of quantum states has been a central problem in quantum information theory due to the impossibility of perfectly distinguish non–orthogonal quantum states. Its implications reflects not only on quantum communication scenarios [GT07, GIN18, Man20], but also in metrology [Gio04, GLM06, GLM11, PLLP19], sensing [DRC17, GRR19], quantum key distribution and cryptography [GRTZ02, LCT14, Ban99, PAB+20, CPP+20].

In the typical *discrimination* scenario [Che00, Ber07, Ber10, BC09], two parties, the transmitter and the receiver, agree on a known shared communication protocol, which defines the (possibly finite) set of quantum states to transmit and discriminate in the best way possible [Hel76b, Hol73a]. A different perspective on the problem has been adopted in recent years, following the growing trend of machine learning studies [Wit14, Gam08, GK10, SSP14, SC02, HHH05, HHH06, SSP14, CDQ+17]. In the context of supervised learning, *classification* problems aim at assigning a sample to one of the

available classes, of which a description is not known, but multiple training samples are provided. Training samples could be copies of the quantum states to classify or other members of the family defining the classes. Classification problems are more general since the description provided by the communication protocol allows the generation of the training samples, therefore expressing a discrimination problem as a classification one. The latter is also more difficult since the classifier has to learn a description or a strategy for the discrimination from the (possibly noisy) training samples, in addition to performing the distinction.

Historically, the discrimination scenario has been investigated the most. Minimum error discrimination has been considered initially for two quantum states [Hel76b, Hol73a], where the optimal solution for the measurement operators assigning the estimate has been given in a closed-form, as shown in detail in Section 2.2.1. Optimality conditions for a bigger set of quantum states have been found [YKL75], but the evaluation of the measurement operators and of the performance usually requires numerical procedures such as semi–definite programming [EMV03]. When the quantum states exhibit symmetries, such evaluation can be further simplified [EF01, EMV04, DPP15]. Despite the advances in the field, in the case of the discrimination of optical states, physical realizations of the optimal receiver end are still an open problem, with the only exception of the Dolinar receiver for the discrimination of two coherent states [Dol73a, Ger04b, TSvLL05a, ADPP11a, CMG07a, CMG07a, BmrcvicvSDcv08]. Along with this scheme, other practical realizations of suboptimal receivers have been proposed for other sets of coherent states [KOSH99, DPL14, MM15, dSGD14, NK18].

Regarding the quantum classification problem, early works frame the same scenario under different names, such as quantum matching (see [SC02] and references within) or quantum state identification [HHH05], and programmable discrimination [BH05]. In the minimum error setting, solutions for the two–classes problem came first for pure states [HHH05, HHH06] and then for general qubit mixed states [GK10, SBCMnT10, FMG19], in the asymptotic and limited–training–samples regimes. Following papers [GK10, SCMTB12, SBCMnT13] have focused on the performance comparison between a joint (collective) measurement strategy involving both the training samples and the one to distinguish, versus an Estimate&Discriminate strategy, where the training copies are used to estimate the classes of states, and the classical information extracted is used to setup the discrimination. The latter strategy results to be suboptimal to the former. The unambiguous version of the classification problem considered in this paper has been addressed in [SZPcv+07, SZBH09], which provided a strategy based on interferometers and photodetectors, that has been demonstrated experimentally [BmrcvicvSDcv08]. Other results on

programmable discriminators can be found in [FDF02, FD04, DB02, BFH06, ZYQ06, HB07, HB08, AH11, Col12, Zho14]. An approach to programmable discriminators for coherent states based on reinforcement learning using passive optical elements, photodetectors and classical adaptive control, has been investigated recently in [BRYC20a].

In the field of quantum optics, discrimination and classification problems have been formulated for the reading of an optical memory. The advantage of using quantum states of light for the discrimination has been established in a series of papers [TEG+08, Llo08, Pir11, PLG+11, LPGM13], while in [SGA15a] the reading has been framed as a classification problem between the vacuum state and a coherent state with unknown parameter, later setup to be a Gaussian a priori distribution around a mean value. The asymptotic behaviour of the collective measurement strategy and the Estimate&Discriminate one has been evaluated and compared, confirming that the former gives better performances. In [SGA15a] it is also conjectured and given some evidence that this holds for *non–Gaussian* Estimate&Discriminate strategies.

2.2.1 The Helstrom problem

We will now consider the following problem:

Problem 2.2.1. An agent gives us an unknown quantum state $\hat{\rho}$. He tells us that this state can be $\hat{\rho}_A$ with probability p_A or $\hat{\rho}_B$ with probability p_B. Our task is to decide optimally whether the state is A or B and estimate the error probability.

The reason why we cannot achieve a zero error probability is due to several aspects that contribute differently:

- The state $\hat{\rho}$, being in a single copy, cannot be reconstructed with certainty, as explained in the former Section 2.1.2.
- Measurements in quantum mechanics are intrinsically probabilistic and destroy the original quantum state.
- States that overlap are more difficult to discriminate because of the nature of quantum measurements.
- One cannot repeat the measurement as many times as he wants because quantum information cannot be copied.

A very important aspect of this problem is the fact that we have full classical knowledge of the states $\hat{\rho}_A$ and $\hat{\rho}_B$. This means that we can choose

our optimal POVM knowing what to expect. We can also prepare in the lab as many copies of these states as we want because we have classical knowledge about them.

2.2.2 Helstrom bound

The Helstrom solution [Hel76a] to this problem is very simple. We have only two possible outcomes, so we aim to build a two-operators POVM. Actually, given the fact that the POVM operators are normalized, we have only one operator as freedom in this problem, because the other one will be constrained by the relation $\hat{E}_B = \hat{\mathbb{1}} - \hat{E}_A$. Our aim is to minimize the error probability, so we will start computing it and writing in a way that makes it more clear how to optimize our discriminator.

$$\begin{aligned} P_{\text{err}} &= \text{tr}[E_B\hat{\rho}_A]p_A + \text{tr}[E_A\hat{\rho}_B]p_B \\ &= p_A + \text{tr}[\hat{E}_A(p_B\hat{\rho}_B - p_A\hat{\rho}_A)] \\ &= p_B - \text{tr}[\hat{E}_B(p_B\hat{\rho}_B - p_A\hat{\rho}_A)] \\ &= \frac{1}{2}(p_A + p_B) - \frac{1}{2}\,\text{tr}[(\hat{E}_B - \hat{E}_A)(p_B\hat{\rho}_B - p_A\hat{\rho}_A)]. \end{aligned} \tag{2.1}$$

We are left with a constant, $p_A + p_B = 1$ and an operator inside the trace, $p_B\hat{\rho}_B - p_A\hat{\rho}_A$. This operator is not positive, but it is still self-adjoint, so it can be diagonalized. In particular, there will exist two subspaces $\mathcal{H}_+ \perp \mathcal{H}_-$ such that

$$p_B\hat{\rho}_B - p_A\hat{\rho}_A = \hat{\rho}_+ - \hat{\rho}_- \qquad \hat{\rho}_\pm \in \mathcal{H}_\pm \qquad \hat{\rho}_\pm \geq 0. \tag{2.2}$$

The construction of these operators is trivial given the spectral decomposition of the original operator. We can simply divide into two classes the eigenvalues, those greater or equal than zero and the others, and group together the corresponding projectors. A similar procedure can be done for the operator $\hat{E}_B - \hat{E}_A$, obtaining the operators $\hat{\theta}_+, \hat{\theta}_-$. Now the trace becomes

$$\text{tr}[\hat{\theta}_+\hat{\rho}_+ + \hat{\theta}_-\hat{\rho}_- - \hat{\theta}_-\hat{\rho}_+ - \hat{\theta}_+\hat{\rho}_-]. \tag{2.3}$$

All the operators $\hat{\rho}_\pm, \hat{\theta}_\pm$ are positive, so we want to maximize the ones with a plus sign in front and minimize the ones with a negative one. We can actually nullify them choosing $\theta_\pm$ as the projector on the space $\mathcal{H}_\pm$. This leads to the formula for the probability of error

$$P_{\text{err}}^{\text{Hel}} = \frac{1}{2}\left(1 - \text{tr}[\hat{\rho}_+ + \hat{\rho}_-]\right)$$
$$= \frac{1}{2}\left(1 - \text{tr}[|p_A\hat{\rho}_A - p_B\hat{\rho}_B|]\right) = \frac{1}{2}\left(1 - \| p_A\hat{\rho}_A - p_B\hat{\rho}_B \|_1\right). \quad (2.4)$$

2.3 The quantum Wasserstein distance of order 1 for qubits

The theory of optimal mass transport [Vil08, Amb08, ABH+13] considers probability distributions on a metric space as distributions of a unit amount of mass. The key element of such theory is the Monge–Kantorovich distance between probability distributions, which is the minimum cost that is required to transport one distribution onto the other, assuming that moving a unit of mass for a unit distance has cost one [Mon81, Kan06, Ver13]. Such distance is also called earth mover's distance or Wasserstein distance of order 1, often shortened to W_1 distance. The exploration of the theory of optimal mass transport has led to the creation of an extremely fruitful field in mathematical analysis, with applications ranging from differential geometry and partial differential equations to machine learning [Amb08, PC19, Ver13].

The most natural distance on the set of the strings of n bits is the Hamming distance [Ham50], which counts the number of different bits. The resulting W_1 distance on the set of the probability distributions on strings of n bits is called Ornstein's $\bar{d}$-distance [Orn73]. Ref. [DPMTL21] proposed a generalization of the W_1 distance to the space of the quantum states of a finite set of qubits, called quantum W_1 distance (or QW_1 in brief). The generalization is based on the notion of neighboring quantum states. Two quantum states of a finite set of qubits are neighboring if they coincide after discarding one qubit. The quantum W_1 distance of Ref. [DPMTL21] is the distance induced by the maximum norm that assigns distance at most 1 to any couple of neighboring states; in the case of quantum states diagonal in the computational basis it recovers Ornstein's $\bar{d}$-distance and inherits most of its properties.

The QW_1 quantity can be computed with a semidefinite program, whose formulation requires to define a notion of Lipschitz constant for quantum observables. The Lipschitz constant of the observable $\hat{H}$ acting on the Hilbert space of n qubits is [DPMTL21]

$$\|\hat{H}\|_L = 2 \max_{i=1,\ldots,n} \min_{\hat{H}_{i^c}} \left\| \hat{H} - \hat{\mathbb{I}}_i \otimes \hat{H}_{i^c} \right\|_\infty , \quad (2.5)$$

where the minimization is performed over all the observables $\hat{H}_{i^c}$ that *do not*

act on the i-th qubit. The quantum W_1 distance between the quantum states $\hat{\rho}$ and $\hat{\sigma}$ can then be expressed as [DPMTL21]

$$\|\hat{\rho} - \hat{\sigma}\|_{W_1} = \max_{\|\hat{H}\|_L \leq 1} \mathrm{Tr}\left[(\hat{\rho} - \hat{\sigma})\,\hat{H}\right] . \tag{2.6}$$

The present paper is based on the following lower bound to the quantum W_1 distance. Let

$$\hat{H}^{(\mathrm{wass})} = \sum_{i=1}^{n} |1\rangle_i\langle 1| \otimes \hat{\mathbb{I}}_{i^c} , \tag{2.7}$$

be the quantum observable that counts the number of ones in the computational basis. We have $\left\|\hat{H}^{(\mathrm{wass})}\right\|_L = 1$ [DPMTL21], therefore for any quantum state $\hat{\rho}$ we have

$$\left\|\hat{\rho} - |0\rangle\langle 0|^{\otimes n}\right\|_{W_1} \geq \mathrm{Tr}\left[\hat{\rho}\,\hat{H}^{(\mathrm{wass})}\right] . \tag{2.8}$$

QW_1 has found several applications in quantum information theory and many-body quantum physics, among which we mention a proof of the equivalence between the microcanonical and the canonical ensembles of quantum statistical mechanics [DPR22] and a proof of limitations of VQA [DPMRF23, CLSS22]. Furthermore, QW_1 has been extended to quantum spin systems on infinite lattices [DPT22]. In the context of quantum state tomography, the quantum W_1 distance has been employed as a quantifier of the quality of the learned quantum state and has led to efficient algorithms to learn Gibbs states of local quantum Hamiltonians [RF21, MPO21, ORFW23]. In the context of quantum machine learning, the quantum W_1 distance has been employed as a cost function of the quantum version of generative adversarial networks [KPM+22, HOR21, AK22, Coy22].

2.3.1 Related approaches

Several quantum generalizations of optimal transport distances have been proposed. One line of research by Carlen, Maas, Datta and Rouzé [CM14, CM17, CM20, RD19, DR20, VVH21, Wir22] defines a quantum Wasserstein distance of order 2 from a Riemannian metric on the space of quantum states based on a quantum analog of a differential structure. Exploiting their quantum differential structure, Refs. [RD19, CM20, GJL20] also define a quantum generalization of the Lipschitz constant and of the Wasserstein distance of order 1. Alternative definitions of quantum Wasserstein distances of order 1 based on a quantum differential structure are proposed in Refs. [CGNT17, RCLO18, CGT18a, CGT18b]. Refs. [Agr13, Agr16, Ike20]

propose quantum Wasserstein distances of order 1 based on a distance between the vectors of the canonical basis.

Another line of research by Golse, Mouhot, Paul and Caglioti [GMP16, CGP21, Gol18, GP17, GP18, CGP20, FECidZ22, CEFŻ21, Duv21, BEŻ22, VVS22] arose in the context of the study of the semiclassical limit of quantum mechanics and defines a family of quantum Wasserstein distances of order 2. Ref. [DPT21] proposes another quantum Wasserstein distance of order 2 where the optimal transport is implemented with quantum channels.

The quantum Wasserstein distance between two quantum states can be defined as the classical Wasserstein distance between the probability distributions of the outcomes of an informationally complete measurement performed on the states, which is a measurement whose probability distribution completely determines the state. This definition has been explored for Gaussian quantum systems with the heterodyne measurement in Refs. [ŻS98, ŻS01, BŻ17].

2.4 Introduction to Quantum Error Correction

Let Q be a quantum register we wish to protect (at least in part) from the action of some external noise source. In a typical QEC scenario [NC00] this problem is addressed through the following three-step procedure:

i) Before the action of the noise, a unitary encoding gate $\hat{V}_{QA}$ is used to distribute the information originally contained in Q on the larger system QA. Here A is an auxiliary quantum register that is assumed to be initialized in a fiduciary quantum state, and that is affected by the same noise that tampers with Q;

ii) After the action of the noise, a measurement on QA is performed to reveal the nature of the latter and, based on the associated outcome, a unitary recovery operation is applied to the system. Equivalently this step can be described by introducing yet an extra quantum register B (also initialized on a fiduciary state but *not* affected by the noise) that is coupled with QA trough a recovering unitary transformation $\hat{W}_{QAB}$ which effectively mimics the measurement and the recovery operation;

iii) The inverse of the gate $\hat{V}_{QA}$ is finally used on QA to refocus the recovered information in Q.

Denoting with $|\psi\rangle_Q$ the input state of Q, the corresponding output state of QA that emerges from the process at the end of the step *iii)* can be expressed

as the density matrix

$$\begin{aligned}\hat{\rho}_{QA}^{(V,W)}(\psi) &:= \mathrm{tr}_B\Big\{\mathcal{V}_{QA}^\dagger \circ \mathcal{W}_{QAB} \circ \Phi_{QA} \\ &\quad \circ \mathcal{V}_{QA}\Big(|\psi\rangle_Q\langle\psi| \otimes |\emptyset\rangle_A\langle\emptyset| \otimes |\emptyset\rangle_B\langle\emptyset|\Big)\Big\} \\ &:= \mathcal{V}_{QA}^\dagger \circ \Phi_{QA}^{(R)} \circ \Phi_{QA} \circ \mathcal{V}_{QA}\Big(|\psi\rangle_Q\langle\psi| \otimes |\emptyset\rangle_A\langle\emptyset|\Big)\end{aligned} \tag{2.9}$$

where $|\emptyset\rangle_X$ represents the fiduciary state of the X register, $\mathrm{tr}_B\{\cdots\}$ is the partial trace over B, and given a unitary $\hat{U}_X$ on X we adopted the symbol $\mathcal{U}_X(\cdots) := \hat{U}_X \cdots \hat{U}_X^\dagger$ to denote its action as super-operator. In the above expressions Φ_{QA} is the LCPT quantum channel [NC00] describing the noise on Q and A, while $\Phi_{QA}^{(R)}(\cdots) := \mathrm{tr}_B\{\mathcal{W}_{QAB}(\cdots \otimes |\emptyset\rangle_B\langle\emptyset|)\}$ is the LCPT (recovery) quantum channel on QA originating from the interaction with B, that attempts to undo the action of Φ_{QA}. An ideal QEC procedure able to completely remove the noise from the system will make sure that $\hat{\rho}_{QA}^{(V,W)}(\psi)$ corresponds to $|\psi\rangle_Q|\emptyset\rangle_A$, irrespectively from the specific choice of $|\psi\rangle_Q$.

2.4.1 Stabilizer codes and the solution for σ noise

In this short section, we will introduce the stabilizer codes [Got97, KLMN01b] to show explicitly the analytic solution for a single σ_i noise acting on some qubits.

Let's consider the group G_n of the Pauli operators

$$G_n = \pm\{\mathbb{1}, \sigma_x, i\sigma_y, \sigma_z\}^{\otimes n} \tag{2.10}$$

and its faithful linear representation $f : G_n \to \mathcal{H}$ of this group acting on an Hilbert space $\mathcal{H}$ of dimension 2^n. It is sufficient to consider this group in order to correct arbitrary kinds of noises, because this group is a basis for the space of operators $\mathcal{H}$, and this means that every Kraus operator acting as noise on our system can be written as a linear combination of elements of the group. We will consider an abelian subgroup S of G_n. All the linear operators in the set $f(S) \subset \mathcal{L}(\mathcal{H})$ are simultaneously diagonalizable because they commute. We call *stabilizer code* of S the linear subspace $\mathcal{H}_S \subset \mathcal{H}$ of eigenvectors with unity eigenvalues. In formulas,

$$|\psi\rangle \in \mathcal{H}_S \iff f(s)\,|\psi\rangle = |\psi\rangle \ \forall s \in S. \tag{2.11}$$

There are some important properties of the G_n group and its representation that are important to remark. The first one is that for each element g of

the group, $g^2 = \pm\mathbb{1}$. The second one is that, for any two elements $g, h \in G_n$, either they commute or they anti-commute. In formulas,

$$gh = \pm hg. \tag{2.12}$$

This property can be used to prove that if the stabilizer S has $n-k$ independent generators, then the stabilizer subspace $\mathcal{H}_S$ has dimension 2^k. Let us now call $\mathcal{E}$ the set of operators we want to correct. Because of property Eq. (2.12), for each element $e_a \in \mathcal{E}$, it is true that, for all $|\psi\rangle \in \mathcal{H}_S, g \in S$,

$$f(g)f(e_a)\,|\psi\rangle = \pm f(e_a)f(g)\,|\psi\rangle = \pm f(e_a)\,|\psi\rangle\,. \tag{2.13}$$

This means that, if we measure one of the operators $f(g), g \in S$, we will obtain the outcome $+1$ if the error commutes with the original operator, -1 if they anticommute. We will now use these properties to show some sufficient condition to have perfect correction, and using them we will recover the known solution to correct a single σ_x qubit error, that will be useful in Chapter 3 and a generic single qubit error, that is a less trivial example that is worth mentioning.

We define the *codewords* as the linear subspace $\mathcal{C} \subset \mathcal{H}$ of logical states that represent the state of the system in a non-minimal subspace. Given a set of errors that we want to correct $\mathcal{E}$, the following holds:

Fact 1. Let $|\psi_i\rangle$ be an orthonormal basis of the codewords $\mathcal{C}$. The errors in $\mathcal{E}$ can be corrected if and only if

$$\langle\psi_i|f(e_a)^\dagger f(e_b)|\psi_j\rangle = C_{ab}\delta_{ij},\ \forall e_a, e_b \in \mathcal{E} \tag{2.14}$$

where δ_{ij} is the Dirac delta.

For the proof of this fact, you can see Ref. [Pre99]. The condition Eq. (2.14) is equivalent to saying that

$$\langle\psi|f(e_a)^\dagger f(e_a)|\psi\rangle = C_{ab} \tag{2.15}$$

is independent of the state $|\psi\rangle$. Using this condition it is straighforward to prove an important fact that defines stabilizer codes:

Fact 2. Let the set of the codewords be equal to the stabilizer eigenspace, $\mathcal{C} = \mathcal{H}_S$. Then, the set of errors $\mathcal{E}$ can be recovered if and only if one of the two following conditions holds:

- $f(e_a)^\dagger f(e_b) \in S$;

- There is an $f(g), g \in S$ that anti-commutes with $f(e_a)^\dagger f(e_b)$;

The proof is the following. In the first case, $\langle\psi|f(e_a)^\dagger f(e_a)|\psi\rangle = 1$, because the operator is in the S subspace, where all the states are eigenstates of the operators $f(S)$ with $+1$ eigenvalue. In the second case,

$$\begin{aligned}\langle\psi|f(e_a)^\dagger f(e_a)|\psi\rangle &= \langle\psi|f(e_a)^\dagger f(e_a)f(g)|\psi\rangle \\ &= -\langle\psi|f(g)f(e_a)^\dagger f(e_a)|\psi\rangle \\ &= -\langle\psi|f(e_a)^\dagger f(e_a)|\psi\rangle = 0.\end{aligned} \tag{2.16}$$

□

We can now use this fact to construct a code that corrects a single error σ_x on a qubit. To do so, we will need n qubits to encode our system, and r qubits to perform the recovery operation. The minimum n to encode a single logical qubit is $n = 3, r = 2$, and this can be understood by looking at the dimension of the subspace $\mathcal{H}_S$. In order to encode a logical qubit, we need at least $\dim(\mathcal{H}_S) \geq 2$, and this cannot be done with $n < 3$. With $n = 3$, we can write explicitly all the elements that we nominated, namely

- The errors that we will correct, in the set $\mathcal{E}$ are

$$\begin{aligned}e_1 &= \sigma_x \otimes \mathbb{1} \otimes \mathbb{1} \\ e_2 &= \mathbb{1} \otimes \sigma_x \otimes \mathbb{1} \\ e_3 &= \mathbb{1} \otimes \mathbb{1} \otimes \sigma_x\end{aligned} \tag{2.17}$$

- The elements of S are

$$\begin{aligned}g_0 &= \mathbb{1}^{\otimes 3} \\ g_1 &= \sigma_x \otimes \sigma_x \otimes \mathbb{1} \\ g_2 &= \mathbb{1} \otimes \sigma_x \otimes \sigma_x \\ g_3 &= \sigma_x \otimes \mathbb{1} \otimes \sigma_x,\end{aligned} \tag{2.18}$$

  and the generators of S can be chosen as g_1, g_2.

- The subspace $\mathcal{H}_S$ has dimension 2 and one of the possible bases is

$$\begin{aligned}|\overline{+}\rangle &= |+\rangle \otimes |+\rangle \otimes |+\rangle \\ |\overline{-}\rangle &= |-\rangle \otimes |-\rangle \otimes |-\rangle\end{aligned} \tag{2.19}$$

- It is trivial to show that the condition in Fact 2 holds because $e_{1,2,3}$ are self-adjoint and the product of any of the two is in S.

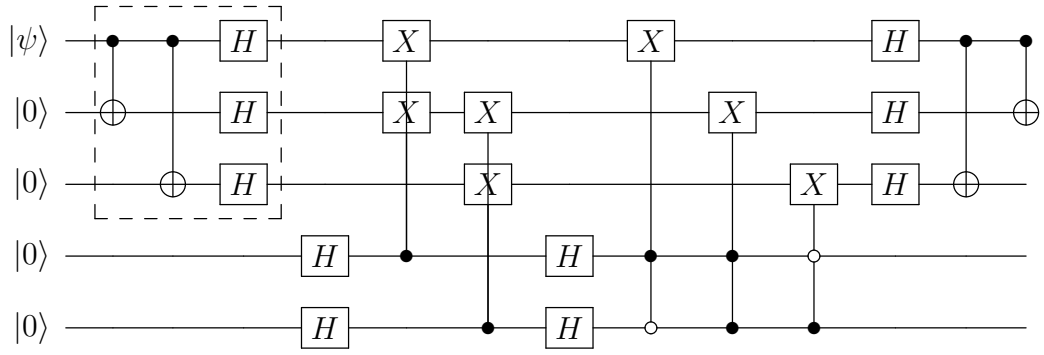


Figure 2.1: stabilizer code for correcting a single σ_x error. The gates between the dashed line represent the encoding of the circuit, where a single qubit gets encoded in a 3-qubit space. The inverse of this block is present at the end to perform decoding. The gates in between represent a coherent detection and correction of the errors that may have happened between the encoding and the correction. The part between the Hadamard is the detection part, where the measurement result is saved on the two recovery qubits. This information is then used with the multi-control σ_x gates to correct the error.

Given all of these conditions, it is very simple to construct the circuit in Fig. 2.1, which can be explained as follows: the input state $|\psi\rangle$ enters the circuit and gets encoded with the trivial encoding

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \to \alpha\,|\overline{+}\rangle + \beta\,|\overline{-}\rangle\,. \tag{2.20}$$

The state then enters the detection phase, where the operators g_1, g_2 are measured and the measurement result is stored in the fourth and fifth qubits respectively. In formulas, if the state $|\psi\rangle$ on the logical qubits enters the detection phase, the transformation done by the fourth qubit (the two are independent) acts with the transformation

$$|\psi\rangle \otimes |0\rangle \to \frac{(|\psi\rangle + M\,|\psi\rangle) \otimes |0\rangle + (|\psi\rangle - M\,|\psi\rangle) \otimes |1\rangle}{2}. \tag{2.21}$$

Given that $M\,|\psi\rangle = \pm\,|\psi\rangle$, and the minus sign is there only if an error happened, the state after this stage is $|\psi\rangle \otimes |0\rangle$ if no error happened, and $|\psi\rangle \otimes |1\rangle$ else. Finally, the control gates perform the correction naturally.

Chapter 3

Improving the speed of variational quantum algorithms for quantum error correction

3.1 Introduction

Performing reliable computations on physically imperfect hardware is something that has become usual nowadays, given the current state of classical computers, which can produce perfect results without any software-side mitigation of the imperfections of the physical media where the computation happens. Error correction is based on the fact that these machines automatically perform, on the hardware side, procedures that allow errors to happen and to be fixed without any intervention from the end user. This kind of setting is even more crucial in a quantum scenario where the current noisy intermediate-scale quantum computers (NISQ) have a much larger error rate than their classical counterparts [LLSK22]. Performing reliable computations with a trustworthy error correction procedure has direct implications not only in quantum computation [Pre12, Pre18], but potentially also in all the other sectors of quantum technology which indirectly relay on it (e.g. quantum communication or quantum key distribution [GRTZ02, LCT14, PAB+20]).

In the typical Quantum Error Correction (QEC) scheme, the quantum information that has to be protected is stored in a subspace of a larger Hilbert space, using an *encoding* procedure. Stabilizer codes [Got97, KLMN01b], also explained briefly in Section 2.4.1, which are within the best analytical results in this field, are not universal because they are tailored for a generic noise acting on a small but unknown subset of qubits. Several attempts have already been made to create a numerical optimization procedure to find an error

correction code for specific noise models [FSW08, KSL08, TKL10, CV20], but these studies are not universal because they rely heavily on the type of noise on the specific quantum circuit and this is a problem because real quantum devices are not characterized by a single kind of quantum noise. Some attempts have been made to characterize the noise of the current and near-term devices [KYG+07, PBJ+15], but these methods will become very difficult to implement soon because classical computers are not able to simulate efficiently quantum circuits when the number of qubits increases. Near-term devices with approximately 50 qubits may already be intractable to simulate for supercomputers [BIS+18].

If we define a figure of merit of the quality of the state after the action of the noise and its corresponding correction, the obvious choice for the kind of maximization algorithm is a Variational Quantum Algorithm (VQA) [CAB+21]. These are hybrid algorithms that couple a quantum computer with a classical one. In this setting, usually, a parametric quantum circuit is applied to some reference state, some measurements are performed on the system, and the outcomes are given to the classical computer to perform a minimization procedure of a given cost function (from this point of view the optimization procedure in a VQA can be seen as the training phase in machine learning). Some examples of this class of algorithms are the variational quantum eigensolver [TCC+21] and the Quantum Approximate Optimization Algorithm [HWO+19]. Proposals to use VQAs to address QEC problems are already present in literature [JRO+17]. Unfortunately, VQAs usually suffer from the phenomenon of barren plateaus [MBS+18, CSV+21], namely the gradient of the cost function decays exponentially with respect to the number of qubits of the system, leading to an untrainable model. The fundamental theoretical reason for such behavior has been associated with the presence of *barren plateaus* which originate when the cost function of the problem is global, i.e. mediated by a highly non-local operator [CSV+21]. To avoid these effects we propose here to guide the VQA search using cost functions inspired to Quantum Wasserstein distance of order 1 (or QW_1 in brief) introduced in Ref. [DPMTL21] as a quantum generalization of the Hamming distance [Ham50] on the set of bit strings. As will detail in the following, at variance with more conventional quantum distances typically adopted in quantum information, QW_1 is lacking a fundamental symmetry (unitary invariance) which makes it a suitable candidate to avoid the barren plateau problem. The rationale behind this is that for unitarily invariant distances as the trace distance or the distances derived from the fidelity, all the states of the computational basis are equally orthogonal and thus have all maximum distance one with respect to the other. The QW_1 functional instead measures how many qubits are different between the two states allowing the

VQA gradient to be less flat in the regions that are not already very close to a local minimum. While this special property of QW_1 has been already observed in other contexts, such as the study of quantum Generative Adversarial Networks presented in [KPM+22, KLM22, HOR21, AK22, Coy22, CYL+19], here we test its effectiveness in the identifying effective QEC procedures. For this purpose, we run a series of numerical tests which compare the performances of a VQA that adopts a conventional (i.e. unitary invariant) cost function, with that of a VQA which instead refers to QW_1-like distances. Our findings confirm that in the second case the effectiveness of the numerical optimization significantly increases both in terms of the probability of a successful training and in the fidelity of the recovered state.

The content of this chapter is organized as follows:

- some basic notions on conventional QEC procedures which allow us to set the notation are presented in the technical background chapter, in Section 2.4;
- a concise, yet rather complete review on the QW_1 distance for qubits is also presented in the technical background, in Section 2.3;
- in Section 3.2 we introduce our VQA discussing the different choices of cost functions that can be used in order to guide it;
- in Section 3.3 we present our numerical results where comparing the performances of the VQA implemented with different types of cost functions.

Lastly, some conclusions are given in Section 3.4.

3.2 Variational Quantum Algorithms

In all this chapter we will follow the convention defined in the technical background, in Section 2.4, denoting as $\hat{\rho}_{QA}^{(V,W)}(\psi)$ the output state of the system after the sequence of encoding, faulty computation, correction and decoding, as computed in Section 2.4. As stated in that section, an ideal QEC procedure able to completely remove the noise from the system will make sure that $\hat{\rho}_{QA}^{(V,W)}(\psi)$ corresponds to $|\psi\rangle_Q|\emptyset\rangle_A$, irrespectively from the specific choice of $|\psi\rangle_Q$. A bona-fide figure of merit to characterize the effectiveness of a generic QEC scheme is hence provided by the average input-output fidelity

$$\overline{F}(V,W) \;:=\; \int d\mu_\psi \; {}_Q\langle\psi|_A\langle\emptyset|\hat{\rho}_{QA}^{(V,W)}(\psi)|\psi\rangle_Q|\emptyset\rangle_A \;, \tag{3.1}$$

where $d\mu_\psi$ is the uniform measure on the set of the input states of Q originated from the Haar measure on the associated unitary group [Vin89] or from an exact or approximate unitary 2-design $\mathcal{S}$ [DCEL09, NC00] that simulates the latter[1]. Notice that by expressing $|\psi\rangle_Q = \hat{U}_Q|\emptyset\rangle_Q$, Section 3.2 can equivalently be casted in the more compact form

$$\overline{F}(V,W) \;=\; {}_{QA}\langle\emptyset|\hat{\rho}_{QA}^{(V,W)}|\emptyset\rangle_{QA}\,, \tag{3.2}$$

with $|\emptyset\rangle_{QA} := |\emptyset\rangle_Q \otimes |\emptyset\rangle_A$ and where the state

$$\begin{aligned}\hat{\rho}_{QA}^{(V,W)} \;:=\; & \frac{1}{|\mathcal{S}|}\sum_{\hat{U}_Q\in\mathcal{S}} \mathcal{U}_Q^\dagger \circ \mathcal{V}_{QA}^\dagger \circ \Phi_{QA}^{(R)} \circ \Phi_{QA} \\ & \circ \mathcal{V}_{QA} \circ \mathcal{U}_Q\Big(|\emptyset\rangle_{QA}\langle\emptyset|\Big)\,,\end{aligned} \tag{3.3}$$

now includes the average over all possible inputs. An ideal QEC procedure will enable one to get $\overline{F}(V,W)=1$. A natural benchmark for lowest admissible $\overline{F}(V,W)$ is represented instead by the value one would get if one decides not to perform corrections on the register that we compute by setting $\hat{V}_{QA}$ and $\hat{W}_{QAB}$ equal to the identity operators i.e.[2]

$$\overline{F}_0 := {}_{QA}\langle\emptyset|\hat{\rho}_{QA}^{(\not\mathbb{1},\not\mathbb{1})}|\emptyset\rangle_{QA}\,. \tag{3.4}$$

While enormous progress has been made in the study of QEC procedures, identifying efficient choices for the operations that lead to (non trivial) high values of $\overline{F}(V,W)$ for a specific noise model, is still a challenging open problem. A possible solution, in this case, is to employ variational quantum algorithms to run numerical searches. Our approach follows a training strategy inspired by the work of Johnson *et al.* [JRO+17]. Assuming hence Q, A, and B to be formed by collections of independent qubits (k for Q, $n-k$ for A, and r for B),

[1] We remind that a unitary 2-design is a probability distribution over the set of unitary operators which can duplicate properties of the probability distribution over the Haar measure for polynomials of degree 2 or less. When Q is a single qubit, a 2-design can be realized by a uniform sampling over a set $\mathcal{S}$ composed by only 6 elements $\hat{\mathbb{1}}$, $\hat{\sigma}_1$, $e^{\pm i\pi/4\hat{\sigma}_1}$, $e^{\pm i\pi/4\hat{\sigma}_2}$ that maps its logical state $|0\rangle_Q$ into the vectors $|0\rangle_Q, |1\rangle_Q, (|0\rangle_Q \pm i\,|1\rangle_Q)/\sqrt{2}, (|0\rangle_Q \mp |1\rangle_Q)/\sqrt{2}$.

[2] Section 3.2 accounts for the noise effects both on Q *and* A. A more conservative estimation of $\overline{F}_0$ can be obtained by focusing directly on the noise on Q alone, i.e. tracing out the A component of $\hat{\rho}_{QA}^{(\not\mathbb{1},\not\mathbb{1})}$ and studying its fidelity with $|\emptyset\rangle_Q$, i.e. $\overline{F}_0^{(\text{strong})} := {}_Q\langle\emptyset|\hat{\rho}_Q^{(\not\mathbb{1},\not\mathbb{1})}|\emptyset\rangle_Q \geq \overline{F}_0$, with $\hat{\rho}_Q^{(\not\mathbb{1},\not\mathbb{1})} := \text{tr}_A\,\hat{\rho}_{QA}^{(\not\mathbb{1},\not\mathbb{1})}$. Notice that for the noise model of Section 3.2.3 the two are directly connected via the identity $\overline{F}_0 = \overline{F}_0^{(\text{strong})} - \frac{n-1}{n}p(1-|\langle 0|\hat{\sigma}|0\rangle|^2)$.

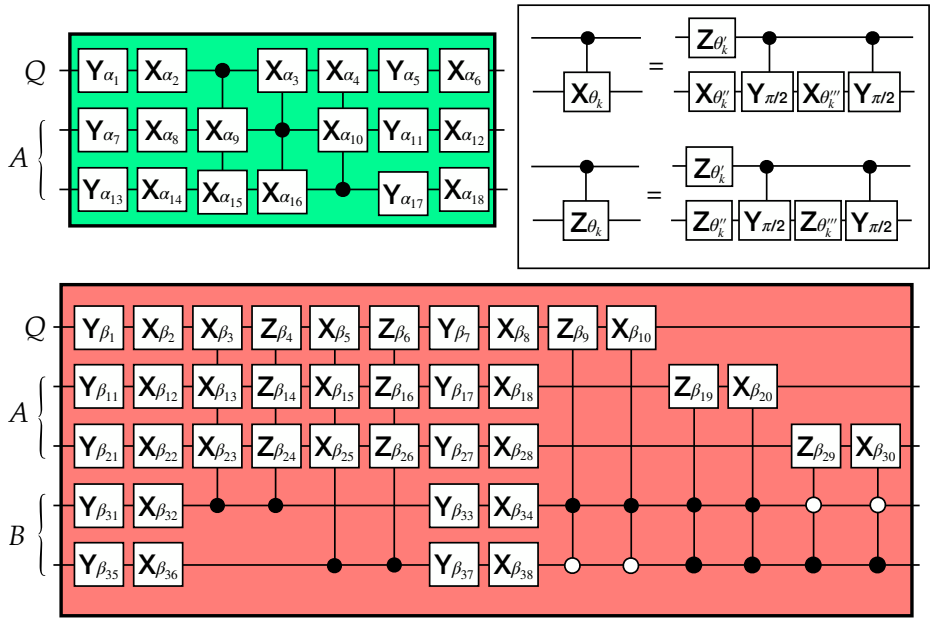


Figure 3.1: Parametric gates $\hat{V}_{QA}(\vec{\alpha})$ (green element) and $\hat{W}_{QAB}(\vec{\beta})$ (red element) used for case of quantum registers Q, A, and B with $k = 1$, $n - k = 2$, and $r = 2$ qubits respectively. Indicating with $\hat{\sigma}_1$, $\hat{\sigma}_2$, and $\hat{\sigma}_3$ the Pauli operators, the X_θ, Y_θ, and Z_θ elements of the figure represent single qubit rotations $e^{-i\theta\hat{\sigma}_1}$, $e^{-i\theta\hat{\sigma}_2}$, and $e^{-i\theta\hat{\sigma}_3}$ with the angles θ determined to the components of the vectors $\vec{\alpha}$, $\vec{\beta}$, respectively. Vertical lines indicate instead quantum control operations which are activated when the control qubits (indicated by the full or empty circles) are in the logical state $|1\rangle$ (full circle) or in $|0\rangle$ (empty circle). As shown on the inset, each one of those gates depend parametrically upon elements of the control vectors $\vec{\alpha}$ and $\vec{\beta}$ through single qubit operations.

we introduce a manifold of transformations $\hat{V}_{QA}(\vec{\alpha})$, $\hat{W}_{QAB}(\vec{\beta})$ parametrized by classical controls vectors $\vec{\alpha}$, $\vec{\beta}$ (see Fig. 3.1), and construct the quantum circuit of Fig. 3.2. The method then proceeds along the following stages:

1. Having selected the values of $\vec{\alpha}$ and $\vec{\beta}$, the register Q is prepared into a collection of known quantum state $\{|\psi(1)\rangle_Q, \cdots, |\psi(m)\rangle_Q\}$ operating on the vector $|\emptyset\rangle_Q = |0\rangle^{\otimes k}$ through action of the control gates $\hat{U}_Q(1), \cdots, \hat{U}_Q(m)$ (first cyan element of the figure) which define the 2-design $\mathcal{S}$ entering in Section 3.2. Each of such inputs is hence evolved via a circuit (pale-orange area of the figure) that emulates both the effect of the noise (patterned square of the figure, see Section 3.2.3 and

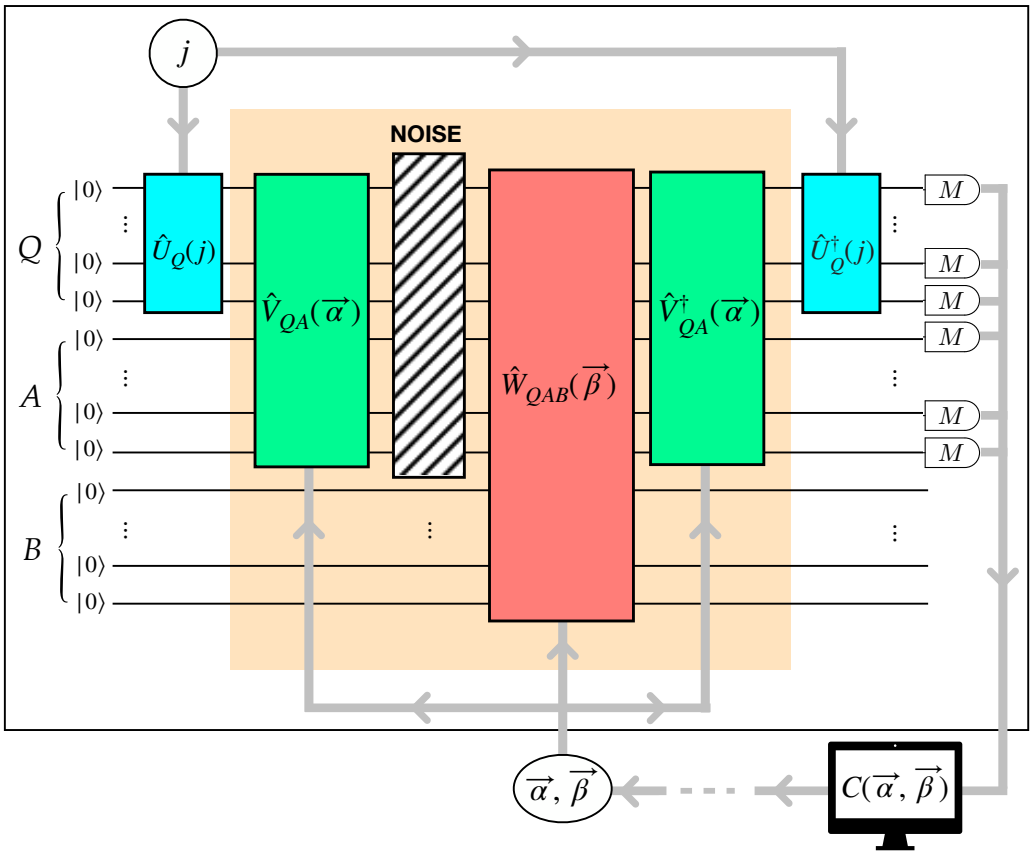


Figure 3.2: Sketch of the variational quantum algorithm: Q, A and B are quantum registers formed respectively by k, $n-k$ and r qubits. The initial information we wish to protect is written in Q by the unitary gate $\hat{U}_Q(j)$ extracted from a 2-design set $\mathcal{S}$; A and B are two auxiliary elements (containing respectively $n-k$ and r qubits) that are used to implement the QEC procedure described by the parametric gates $\hat{V}_{QA}(\vec{\alpha})$, $\hat{W}_{QAB}(\vec{\beta})$, and $\hat{V}^\dagger_{QA}(\vec{\alpha})$ of Fig. 3.1. The patterned element in the central part of the scheme represents the noise on Q and A (no noise is assumed to be active on B). Lastly, the D-shaped measurements at the end of the circuit represent local measurements on QA whose outcomes over the entire collection of the possible inputs generated by the entire set $\mathcal{S}$, are processed by a classical computer which, evaluating the cost function $C(\vec{\alpha}, \vec{\beta})$ defined in Section 3.2.2, decides how to update the values of the parameters $\vec{\alpha}$ and $\vec{\beta}$. Thick grey lines in the figure represent classical control lines.

Fig. 3.4), and the transformations $\hat{V}_{QA}(\vec{\alpha})$, $\hat{W}_{QAB}(\vec{\beta})$, and $\hat{V}^{\dagger}_{QA}(\vec{\alpha})$ that are meant to implement the steps *ii)* and *iii)* of the QEC procedure (green and red elements of the figure). Notice that in the ideal case (i.e. if $\hat{V}_{QA}(\vec{\alpha})$ and $\hat{W}_{QAB}(\vec{\beta})$ manage to completely suppress the noise) then in correspondence with the input $|\psi(j)\rangle_Q$ the registers QA should emerge in the state $|\psi(j)\rangle_Q \otimes |\emptyset\rangle_A := |\psi(j)\rangle_Q \otimes |0\rangle^{\otimes n-k}$, which will be hence mapped into the final configuration $|\emptyset\rangle_{QA} := |0\rangle^{\otimes n}$ by the inverse $\hat{U}^{\dagger}_Q(j)$ of the state preparation gate (second cyan element of the figure).

2. For each choice of the index $j \in \{1, \cdots, m\}$ a measurement on the system is performed at the end of the transformations described in stage 1 and the resulting m collected outcomes used to compute a cost function $C(\vec{\alpha}, \vec{\beta})$ which evaluates the effectiveness of the adopted QEC strategy in leading large values of the average input-output fidelity. The specific choice of the cost function is very important and is discussed in Section 3.2.1.

3. A classical computer decides, given the results of the measurement, how to change the value of the parameters $\vec{\alpha}$ and $\vec{\beta}$ to be used in the subsequent run in order to minimize the cost function $C(\vec{\alpha}, \vec{\beta})$. This is discussed in detail in Section 3.2.2.

3.2.1 Cost function

The natural choice for the cost function at the stage 2 of our algorithm is provided by the expectation value of the self-adjoint operator

$$\hat{H}^{(\mathrm{fid})}_{QA} := \hat{\mathbb{1}}_{QA} - |\emptyset\rangle_{QA}\langle\emptyset| \ , \tag{3.5}$$

computed on the mean state of system QA which emerges at the output of the quantum circuit of Fig. 3.2, i.e. the quantity

$$C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta}) \ := \ \mathrm{tr}\{\hat{\rho}^{(V(\vec{\alpha}),W(\vec{\beta}))}_{QA} \hat{H}^{(\mathrm{fid})}_{QA}\} \ , \tag{3.6}$$

where $\hat{\rho}^{(V(\vec{\alpha}),W(\vec{\beta}))}_{QA}$ is the density matrix (Section 3.2) evaluated for $\hat{V}_{QA} = \hat{V}_{QA}(\vec{\alpha})$ and $\hat{W}_{QAB} = \hat{W}_{QAB}(\vec{\beta})$. This choice has two main advantages. First of all, the expectation value $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta})$ can be evaluated by performing (simple) local measurement on the qubits of Q and A (indeed it can be computed by simply checking whether of not each one of them is in the logical state $|0\rangle$). Most importantly, since by explicit evaluation one has that $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta}) = 1 - \overline{F}(V(\vec{\alpha}), W(\vec{\beta}))$, it is clear that by using (Section 3.2.1) the

algorithm will be forced to look for values of $\vec{\alpha}$, $\vec{\beta}$ that yield higher average input-output fidelities. Despite all this, the use of $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$ as a cost function, has a major drawback associated with the fact that the spectrum of the Hamiltonian $\hat{H}^{(\mathrm{fid})}_{QA}$ exhibits maximum degeneracy with respect to space orthogonal to the target state $|\emptyset\rangle_{QA}$ (see Fig. 3.3). Due to this fact a numerical search based on a training procedure that simply target the minimization of $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$, has non trivial chances to get stuck somewhere in the large flat plateau associated with the eigenvalue 1 of $\hat{H}^{(\mathrm{fid})}_{QA}$ without finding any good direction. in the large flat plateau A possible way to avoid this problem is to introduce new cost-functions Hamiltonians which, while maintaining the target vector $|\emptyset\rangle_{QA}$ as a unique ground state and still being easy to compute, manage to remove the huge degeneracy of the excited part of the spectra of $\hat{H}^{(\mathrm{fid})}_{QA}$. Our choice is based on the quantum Wasserstein distance of order 1 (W_1) introduced Ref. [DPMTL21] which, even though it lacks some interesting properties that the fidelity has, is less likely to be affected by the barren plateaus phenomena [CSV+21]. As mentioned in Section 2.3 good estimation of the W_1 distance that separate $\hat{\rho}^{(V(\vec{\alpha}),W(\vec{\beta}))}_{QA}$ from the target state, is provided by the following quantity

$$C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta}) := \mathrm{tr}\{\hat{\rho}^{(V(\vec{\alpha}),W(\vec{\beta}))}_{QA}\hat{H}^{(\mathrm{wass})}_{QA}\}\,, \tag{3.7}$$

$$\hat{H}^{(\mathrm{wass})}_{QA} := \sum_{j=1}^{n} j\;\hat{\Pi}^{(j)}_{QA}\,, \tag{3.8}$$

where $\hat{H}^{(\mathrm{wass})}_{QA}$ is the Hamiltonian in Eq. (2.7) which we express here in terms of the projectors $\hat{\Pi}^{(j)}_{QA}$ on the sub-space of the register QA in which we have j qubits in $|1\rangle$ and the remaining one in $|0\rangle$. Observe that, as already anticipated, $\hat{H}^{(\mathrm{wass})}_{QA}$ is the sum of the number operators acting on the individual qubits of the register QA as in (2.7): accordingly, as $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$, $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ can be computed from local measurement. What $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ does is to count the total number of logical ones present in the system. To understand why using (3.7) could in principle lead to a more efficient numerical search than the one obtained by using (3.6), notice that Section 3.2.1 can be equivalently written as $\hat{H}^{(\mathrm{fid})}_{QA} = \sum_{j=1}^{n}\hat{\Pi}^{(j)}_{QA}$. A comparison with (3.8) reveals hence that indeed while both $\hat{H}^{(\mathrm{fid})}_{QA}$ and $\hat{H}^{(\mathrm{wass})}_{QA}$ admit $|\emptyset\rangle_{QA}$ as a unique ground state, the Wasserstein Hamiltonian removes large part of the degeneracy of the high energy spectrum of the fidelity Hamiltonian. Accordingly, it is reasonable to expect that a numerical search that uses $\hat{H}^{(\mathrm{wass})}_{QA}$, has fewer chances to get

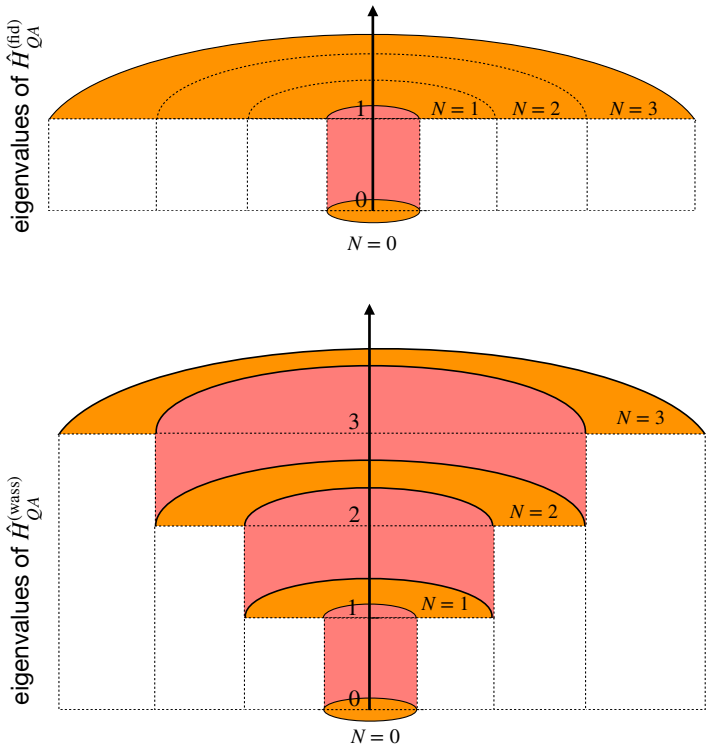


Figure 3.3: Pictorial rendering of the spectra of the Hamiltonians $\hat{H}_{QA}^{(\mathrm{fid})}$ (top panel) and $\hat{H}_{QA}^{(\mathrm{wass})}$ (lower panel). While $\hat{H}_{QA}^{(\mathrm{fid})}$ is characterized by a unique, flat plateau that includes all the excited state, $\hat{H}_{QA}^{(\mathrm{wass})}$ partially removes the associated degeneracy assigning higher energy to subspaces that have higher number of qubits in the logical state $|1\rangle$.

trapped into regions of constant energy (barren plateau) than a search based on $\hat{H}_{QA}^{(\mathrm{fid})}$,[3].

[3]It goes without mentioning that alternative choices for the cost function Hamiltonians are also available. For instance, one can use operators that also remove the residual degeneracies that affect $\hat{H}_{QA}^{(\mathrm{wass})}$ – e.g. using the operator $\hat{H}_{QA}^{(\mathrm{full})} = \sum_{\ell=1}^{n} w_\ell \hat{\pi}_\ell$ with ω_ℓ positive weights selected so that different allocation of $|1\rangle$ states inside the eigenspaces of $\hat{H}_{QA}^{(\mathrm{wass})}$ get an assigned ordering. Our numerical analysis however seems to indicate that these refinements do not contribute significantly in improving numerical search of the algorithm.

3.2.2 Descent algorithm

The algorithm that we used for this work is a gradient descent algorithm with momentum [NW06]. To overcome the numerical difficulties of using finite differences to estimate the gradients of the cost function $C(\vec{\alpha},\vec{\beta})$, we exploit a variation of the parameter-shift rule introduced in [SBG+19] which reduces the problem to compute linear combinations of the function itself evaluated in different points that are not infinitesimally close. Specifically, we observe that, irrespectively from the choice of the operator $\hat{H}_{QA}$, the functional dependence of $C(\vec{\alpha},\vec{\beta})$ upon the j-th component of the vector $\vec{\beta}$ is of the form

$$C(\vec{\alpha},\vec{\beta}) = f(\beta_j) := \sum_k \mathrm{tr}\left\{\hat{\Omega}_1^{(k)} e^{i\beta_j\hat{\sigma}}\hat{\Omega}_2^{(k)} e^{-i\beta_j\hat{\sigma}}\right\}, \tag{3.9}$$

with $\hat{\Omega}_{1,2}^{(k)}$ being multi-qubits operators which do not depend upon β_j, and with $e^{-i\beta_j\hat{\sigma}}$ a single qubit rotation generated by an element $\hat{\sigma}$ of the Pauli set. Therefore its gradient can be written as

$$\begin{aligned}\frac{\partial C(\vec{\alpha},\vec{\beta})}{\partial\beta_j} &= i\sum_k \mathrm{tr}\left\{\hat{\Omega}_1^{(k)} e^{i\beta_j\hat{\sigma}}[\hat{\sigma},\hat{\Omega}_2^{(k)}]e^{-i\beta_j\hat{\sigma}}\right\} \\ &= f(\beta_j+\tfrac{\pi}{4}) - f(\beta_j-\tfrac{\pi}{4})\,,\end{aligned} \tag{3.10}$$

where in the last passage we used the identity

$$i[\hat{\sigma},\hat{\Omega}_2^{(k)}] = e^{i\frac{\pi}{4}\hat{\sigma}}\hat{\Omega}_2^{(k)} e^{-i\frac{\pi}{4}\hat{\sigma}} - e^{-i\frac{\pi}{4}\hat{\sigma}}\hat{\Omega}_2^{(k)} e^{i\frac{\pi}{4}\hat{\sigma}}. \tag{3.11}$$

The gradient with respect the vector $\vec{\alpha}$ can be computed similarly. In this case however we observe that, due to the fact that $\hat{\rho}_{QA}^{(V(\vec{\alpha}),W(\vec{\beta}))}(\psi)$ depends upon the parameters $\vec{\alpha}$ via $\hat{V}_{QA}(\vec{\alpha})$ and through its adjoint $\hat{V}_{QA}^{\dagger}(\vec{\alpha})$, the dependence of $C(\vec{\alpha},\vec{\beta})$ upon the j-th component of $\vec{\alpha}$ is slightly more complex. Indeed in this case we have

$$C(\vec{\alpha},\vec{\beta}) = g(\alpha_j,\alpha_j)\,, \tag{3.12}$$

where $g(\alpha_j^{(1)},\alpha_j^{(2)})$ is the function

$$\begin{aligned}g(\alpha_j^{(1)},\alpha_j^{(2)}) := \sum_k \quad & \mathrm{tr}\left\{\hat{\Omega}_1^{(k)} e^{i\alpha_j^{(1)}\hat{\sigma}}\hat{\Omega}_2^{(k)} e^{-i\alpha_j^{(1)}\hat{\sigma}}\right. \\ & \left.\times\hat{\Omega}_3^{(k)} e^{i\alpha_j^{(2)}\hat{\sigma}}\hat{\Omega}_4^{(k)} e^{-i\alpha_j^{(2)}\hat{\sigma}}\right\}\,,\end{aligned} \tag{3.13}$$

with $\hat{\Omega}^{(k)}_{1,2,3,4}$ representing multi-qubits operators which do not depend neither upon $\alpha^{(1)}_j$ nor $\alpha^{(2)}_j$. It is important to stress that $g(\alpha^{(1)}_j, \alpha^{(2)}_j)$ can be computed using the same circuit of Fig. 3.2, by simply replacing the phases α_j of $\hat{V}_{QA}(\vec{\alpha})$ and $\hat{V}^\dagger_{QA}(\vec{\alpha})$ with $\alpha^{(1)}_j$ and $\alpha^{(2)}_j$ respectively. Notice finally that exploiting the identity Eq. (3.11) we can write

$$\begin{aligned} \frac{\partial C(\vec{\alpha},\vec{\beta})}{\partial \alpha_j} &= \frac{\partial g(\alpha^{(1)}_j, \alpha_j)}{\partial \alpha^{(1)}_j}\bigg|_{\alpha^{(1)}_j=\alpha_j} + \frac{\partial g(\alpha_j, \alpha^{(2)}_j)}{\partial \alpha^{(2)}_j}\bigg|_{\alpha^{(2)}_j=\alpha_j} \\ &= g(\alpha_j + \tfrac{\pi}{4}, \alpha_j) - g(\alpha_j - \tfrac{\pi}{4}, \alpha_j) \\ &+ g(\alpha_j, \alpha_j + \tfrac{\pi}{4}) - g(\alpha_j, \alpha_j - \tfrac{\pi}{4}) \ , \end{aligned} \tag{3.14}$$

which shows that computing the gradient of $C(\vec{\alpha},\vec{\beta})$ with respect to α_j simply accounts to evaluate the circuit that express $g(\alpha^{(1)}_j, \alpha^{(2)}_j)$ for four distinct values of the parameters.

3.2.3 Noise model

The scheme presented so far can in principle be applied to arbitrary classes of noises. In our research however we focused on a specific model that has been extensively studied in the literature producing explicit examples of efficient QEC solutions which can be used as a theoretical benchmark for our variational search. Specifically we assume Q and A to be respectively a single qubit register ($k = 1$) and a two qubit register ($n = 3$), globally affected by a given species of single-qubit noise [Got09, KLMN01a]. These transformations can be represented in terms of a LCPT map of the form

$$\Phi_{QA}(\cdots) = \sum_{\ell=0}^{n} \hat{K}^{(\ell)}_{QA} \cdots \hat{K}^{(\ell)\dagger}_{QA} \ , \tag{3.15}$$

with Kraus operators [NC00]

$$\hat{K}^{(0)}_{QA} := \sqrt{1-p}\ \hat{\mathbb{1}}_{QA} \ , \qquad \hat{K}^{(\ell)}_{QA} := \sqrt{\frac{p}{n}}\ \hat{\sigma}^{(\ell)} \ , \tag{3.16}$$

where for $\ell \in \{1, \cdots, n\}$, $\hat{\sigma}^{(\ell)}$ is the Pauli operator acting on the ℓ-th qubit of QA which defines the noise species we have selected. For instance, in the case we choose to describe phase-flip noise then $\hat{\sigma}^{(\ell)} = \hat{\sigma}^{(\ell)}_3$, while for describing bit-flip we have $\hat{\sigma}^{(\ell)} = \hat{\sigma}^{(\ell)}_1$. Explicit examples of $\hat{V}_{QA}$, $\hat{W}_{QAB}$ which allow for exact suppression of the noise ($\overline{F}(V, W) = 1$) are shown in Fig. 3.5. Notice that by construction the circuit parametrization of $\hat{V}_{QA}(\vec{\alpha}), \hat{W}_{QAB}(\vec{\beta})$ given in

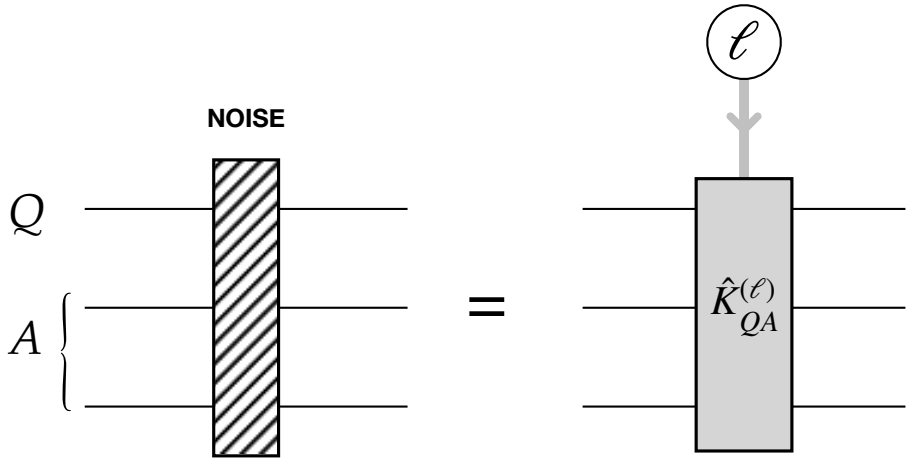


Figure 3.4: Circuital implementation of the noise element of Fig. 3.2: here $\hat{K}^{(\ell)}_{QA}$ are weighted unitaries of Eq. (3.16).

Fig. 3.2 include such gates as special solution: accordingly if properly guided by an efficient cost function, our numerical VQA search has a chance to find the solution of Fig. 3.5.

3.3 Results

In this section we study the impact of the cost function on the efficiency of the optimization algorithm of Section 3.2. Assuming the single-qubit noise model detailed in Section 3.2.3 and taking B to be a $r = 2$ qubit register, we run two distinct numerical searches: the first obtained by identifying $C(\vec{\alpha}, \vec{\beta})$ with $C^{(\text{fid})}(\vec{\alpha}, \vec{\beta})$ and the second choosing instead $C^{(\text{wass})}(\vec{\alpha}, \vec{\beta})$. Results are reported in Figs. 3.6 and 3.7 for two different choices of the noise models (3.15), i.e. phase-flip and bit-flip. For both, we compare the input-output average fidelity (3.2) at the end of the procedure obtained with the two different cost functions, and the number of iterations M needed for convergence. Regarding this last quantity we set a maximum value M_{max} equal to 2000 before convergence and we chose this limit mainly with practical choices like the maximum time for the simulation, enforcing that a single run does not require more than a few hours of computational time: in case the algorithm fails to reach the convergency we simply stop the numerical search (this is the reason for the peak at the end of the upper orange plot in Fig. 3.7). The plots report only the simulations that manage to achieve an average fidelity

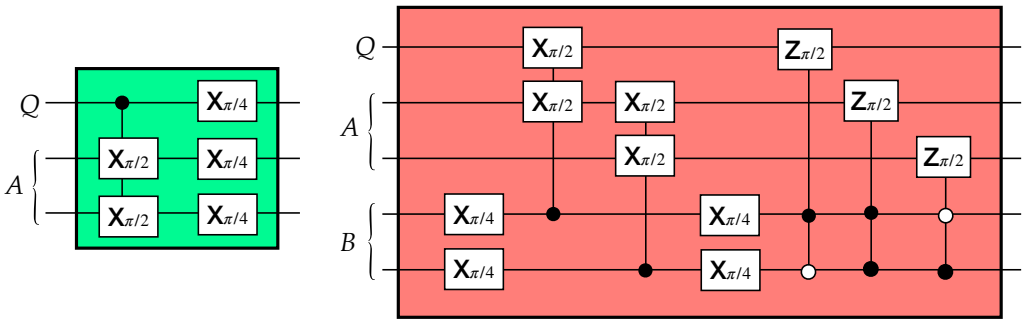


Figure 3.5: Circuital implementations of the ideal transformations $\hat{V}_{QA}(\vec{\alpha})$ (left) and $\hat{W}_{QAB}(\vec{\beta})$ (right) which allow for exact noise suppression of a single-qubit bit-flip noise model [i.e. Section 3.2.3 with $\hat{\sigma}^{(\ell)} = \hat{\sigma}_1^{(\ell)}$] using a quantum register B with $r = 2$ qubit. gates.

that is greater or equal than no-correction threshold bound $\overline{F}_0$.

The first thing to observe is that for both noise models, $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta})$ has problem in reaching the do-nothing threshold $\overline{F}_0$: the probability of success being 2.6% for the phase-flip case of Fig. 3.7 and only 0.2% for the bit-flip case of Fig. 3.6 (for both noise models the total number of simulations analyzed was 500). Observe also that in this last case the algorithm never yields average input-output fidelity values strictly larger than $\overline{F}_0$ and that, even in those cases, it requires a number M of iterations which saturate the maximum allowed value $M_{\max}$ (blue peak in the upper plot of Fig. 3.7). $C^{(\mathrm{was})}(\vec{\alpha}, \vec{\beta})$ performs definitely better: to begin with it succeeds in overcoming the threshold $\overline{F}_0$ in one third of the simulations (specifically 40.6% for the phase-flip noise model and 29.6% for the bit-flip noise model). Furthermore, the algorithm reaches convergency with a number of iterations that are typically smaller than those required by $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta})$.

To better enlighten the differences between the two cost functions, we proceeded with further simulations, whose results are summarized in Fig. 3.8. The idea here is to run a two-step optimization process composed by two sequences of runs: in the first run we start the optimization procedure from a random point in the parameter space $(\vec{\alpha}, \vec{\beta})$ with one of the two cost functions (say $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta})$), up to convergence; after that we start a second optimization run using the other cost function (say $C^{(\mathrm{wass})}(\vec{\alpha}, \vec{\beta})$) but assuming as initial condition for the parameters the final point reached by the first run. The plots report the difference in fidelity between the second and the first run: when we start using the $C^{(\mathrm{wass})}(\vec{\alpha}, \vec{\beta})$ in the first run, the fidelity cannot further improve the result that is already found, and this is represented by the fact

that the best improvement is of the order of 10^{-5}; on the contrary, if we started employing $C^{(\mathrm{fid})}(\vec{\alpha}, \vec{\beta})$ in the first run, the use of $C^{(\mathrm{wass})}(\vec{\alpha}, \vec{\beta})$ in the second run typically yields substantial improvements of the performance[4]. Moreover, we sampled some single descent processes and plotted the cost in function of the iteration. When we move from fidelity to W_1, the descent part after the change of cost function is qualitatively indistinguishable from starting from a random point.

[4]It has to be said that in few cases the figure of merit is worse after the second optimization – see the negative bar in right panel of Fig. 3.8. This is due to the fact that when using $C^{(\mathrm{wass})}(\vec{\alpha}, \vec{\beta})$ we are not maximizing the fidelity but minimizing a function whose stationary point corresponds to the maximum of the latter: accordingly the final point of convergence for $C^{(\mathrm{wass})}(\vec{\alpha}, \vec{\beta})$ can be slightly off mark in terms of fidelity. This is not a problem because these two functions do not have a constant ratio, and we checked that the inequalities between them are still satisfied.

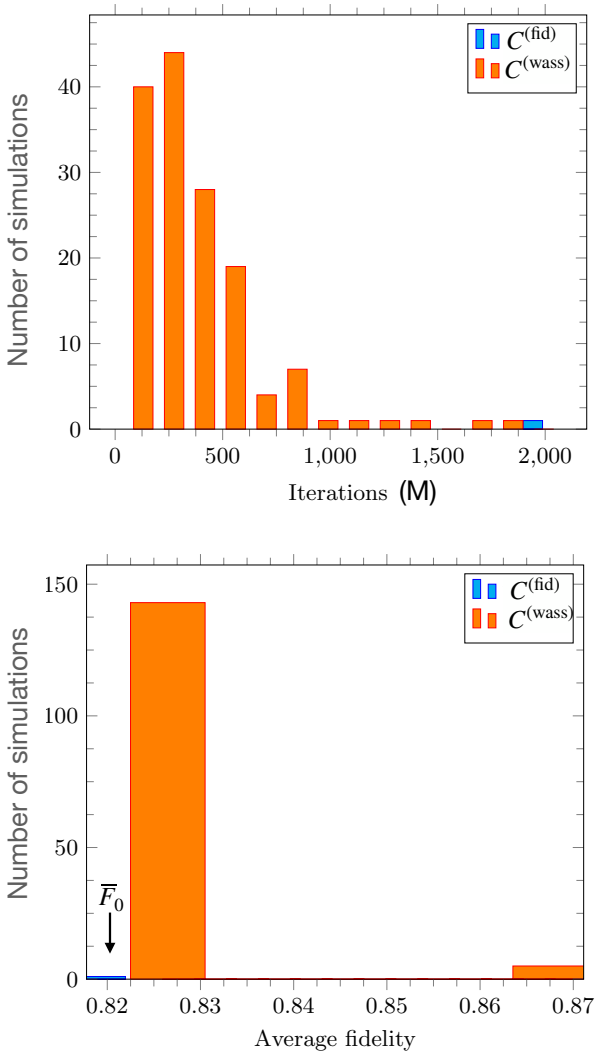


Figure 3.6: Comparison of the input-output average fidelity (3.2) attainable by running our optimization algorithm using the cost function $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$ (blue data) and $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ (orange data). Here the error model is a single-qubit bit-flip noise ($\hat{\sigma}=\hat{\sigma}_1$ in Section 3.2.3) with $p=0.8$. The no error correction threshold (3.4) of this scheme is $\overline{F}_0\approx 0.822$ – orange peak in the fidelity plot, up to numerical precision. Only the runs that produced a fidelity of at least $\overline{F}_0$ have been included: for $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$ this happens in the 0.2% of the runs, while for $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ for the 29.6%.

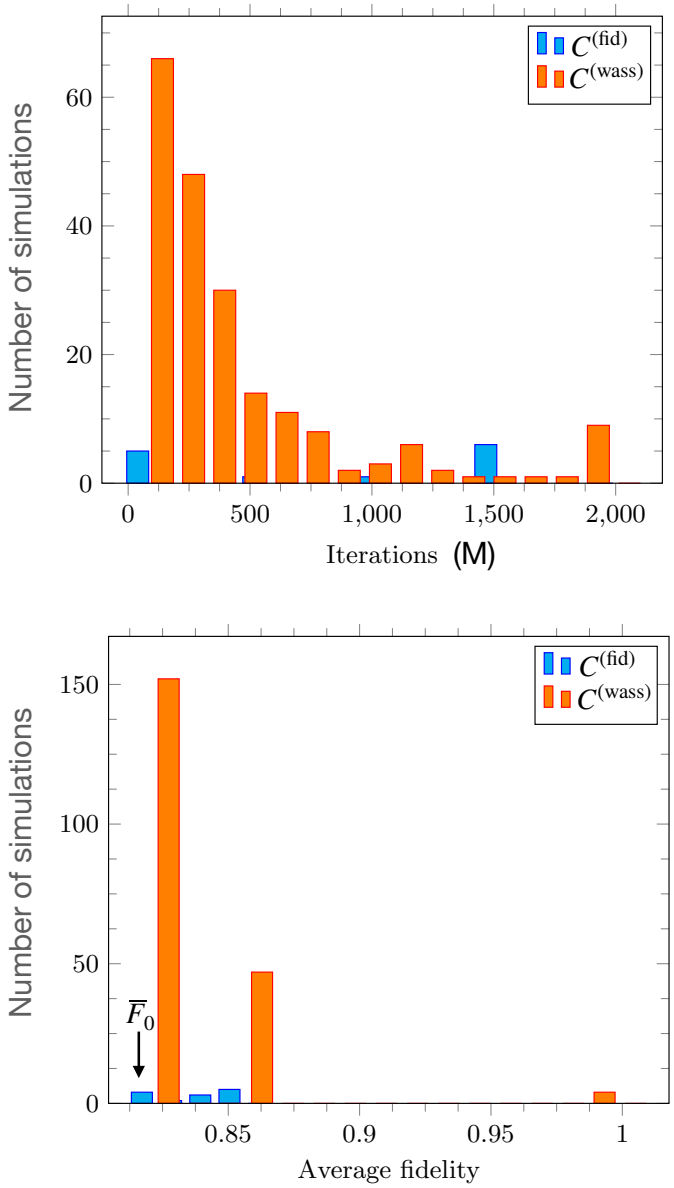


Figure 3.7: Comparison of the the input-output average fidelity (3.2) attainable by running our optimization algorithm using the cost function $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$ (blue data) and $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ (orange data). Here the error model is a single-qubit phase-flip noise ($\hat{\sigma}=\hat{\sigma}_3$ in Section 3.2.3 with $p=0.8$. The no error correction threshold (3.4) of this scheme is $\overline{F}_0\approx 0.822$ – orange peak in the fidelity plot, up to numerical precision. Only the runs that produced a fidelity of at least $\overline{F}_0$ have been included: for $C^{(\mathrm{fid})}(\vec{\alpha},\vec{\beta})$ this corresponds to the 2.6% of the runs, while for $C^{(\mathrm{wass})}(\vec{\alpha},\vec{\beta})$ the success probability is 40.6%.

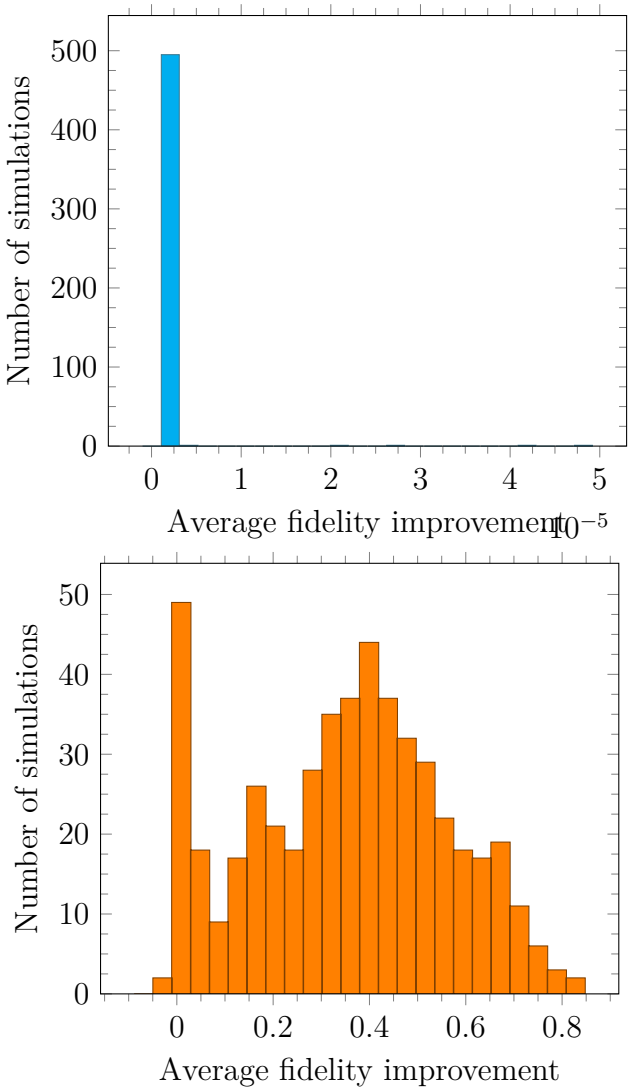


Figure 3.8: Improvement of simulations when changing the cost function in a two run optimization process that uses different cost functions to drive the descent algorithm. In the left plot, we started the descent on a random initial point, ran the optimization using $C^{(\text{wass})}(\vec{\alpha}, \vec{\beta})$ as cost function until convergence, and then we started the descent algorithm again but using $C^{(\text{fid})}(\vec{\alpha}, \vec{\beta})$ as cost function, starting from the final point of the previous descent. In the right part, the roles of the two cost functions are inverted (we start using $C^{(\text{fid})}(\vec{\alpha}, \vec{\beta})$ and then we use $C^{(\text{wass})}(\vec{\alpha}, \vec{\beta})$). The histograms represent the difference in average input-output fidelity (3.1) after the change of cost function, namely the difference between the fidelity achieved after the second descent and the fidelity after the first descent (positive values correspond to improved performances).Please notice the scale difference on the x-axis between the left and right plot.

3.4 Conclusions

To summarize, we have shown a variational quantum algorithm that allows finding the most suitable error correction procedure for a specific noise on quantum hardware. We compared the performance of two different versions of this algorithm using two different cost functions, the fidelity and an approximation of the quantum Wasserstein distance of order one. We compared the difference in speed and the ability to obtain a useful solution between the two algorithms, finding really different trends between the two optimization procedures. The optimization process based on the fidelity suffers greatly from the phenomenon of the barren plateaus, leading to very slow convergence or no convergence at all, while the algorithm based on the quantum W_1 distance allows us to find the configurations that correct the errors in the examples that we explored. The obtained results show a clear improvement and allow us to explore further improvements of these methods, as using different algorithms for the minimization process, e.g. stochastic gradient descent or higher-order algorithms like Newton or pseudo-Newton algorithms.

Given that the gradient can be expressed only with the cost function evaluated in a small number of circuits that differ only in the parameter choice, the gradient of the cost function can be computed on the same hardware that will be used for the correction procedure. Moreover, simulating this circuit may be difficult because of the exponential scaling of the dimension of the Hilbert space of a set of qubits, but this problem does not apply when all the circuit is built on hardware, gaining a quantum advantage. For the same reason, the same procedure can be iterated to compute the exact Hessian of the cost function and then apply a second-order method like the Newton method as a descent algorithm. However, this has not been done because the circuits that we marked as useful have a relatively big number of parameters, and computing the hessian scales quadratically with this number, leading to intractable computations.

With this work, we have shown a clear advantage in the use of this QW_1 distance approximation in a gradient-based optimization algorithm. In future work, it may be interesting to study the effect of this cost function also on gradient-free optimization algorithms, that do also suffer from the barren-plateaus phenomenon, as shown in Ref. [ACC+21].

Chapter 4

An agnostic-Dolinar receiver for coherent states classification

4.1 Introduction

Following the introduction in Section 2.2, discrimination of quantum states, particularly non-orthogonal ones, has been a key challenge in quantum information theory, influencing areas such as quantum communication, metrology, and cryptography. Traditionally, the focus has been on minimizing errors in discriminating between a set number of states, with optimal solutions available for specific cases. However, the emergence of machine learning has shifted some attention towards quantum classification, which generalizes the discrimination problem by requiring the classifier to learn and distinguish between classes based on training samples. While joint measurement strategies generally outperform estimate-and-discriminate approaches, physical implementations of optimal receivers, especially in optical systems, remain largely unresolved, with a few exceptions. In summary, many problems in this field can be categorized into two types: Minimum Error Discrimination and Minimum Error Classification of quantum states. The former, a core problem in Quantum Information Theory, relies on having classical knowledge of the quantum states to be discriminated. The latter, emerging from recent advances in machine learning, replaces this classical description with the availability of multiple training copies corresponding to the classes of quantum states to be distinguished.

In this chapter We consider the simplest scenario concerning the classification of an unknown state belonging to one of two classes of coherent states that are assigned by giving access to a certain number of training copies. A practical scenario where this task could be relevant is an optical

link through a stochastic channel where the attenuation is so unpredictable and random that over sufficiently long time intervals, the signal intensity can be assumed to be completely unknown, and the same holds for the added phase. In this context one may try to exploit the existence of stability periods in the perturbations induced by the noise, to set communication protocols that consist in sending samples of the two types of training signals followed by the quantum state to classify. To begin with we show that via simple linear optics, the problem we are facing can always be reduced to the special symmetric case where the two classes of inputs differs only by the sign of the associated coherent amplitudes. We hence evaluate the optimal bound for the probability of success of the classification task, under the assumption that the protocol to use is phase invariant. Such threshold can be explicitly calculated for any number of input copies of the training states, and for any given prior distribution of the coherent amplitudes that characterize them. Unforturnately the POVM measurement that ensures the attainability of the optimal bound relays on highly non-linear optical processes that are not feasible with current technology. In alternative we propose a modification of the conventional Dolinar receiver [Dol73a] that we dub agnostic-Dolinar receiver, whose implementation is instead at reach with conventional quantum optical procedures. While being sub-optimal when employed with a finite number n of training copies, the proposed scheme it is shown to saturate to the optimal bound in the asymptotic limit $n \to \infty$. Most importantly, for all n, it yields a clear advantage when compared with respect to simple Estimate&Discriminate strategies that involve estimations performed on a fraction of the training samples.

The chapter is organized as follows. In Section 4.2 we introduce the problem and review the original Dolinar receiver. Section 4.3 is the main section of the manuscript and includes all the original results of our work. Here we reduce the problem to a symmetric scenario, provide an optimal bound for the problem, and present our apparatus. We study its performance by comparing it with an Estimate&Discriminate strategy based on a miscalibrated Dolinar scheme, and also compute the optimal error probability of the problem under different assumptions on the prior information available.

4.2 Discrimination and Classification of coherent optical signals

This section is dedicated to set the problem, introduce the notation, and review some basic facts.

4.2.1 Discrimination vs Classification

The discrimination and classification of quantum states are two distinct primitives of quantum information processing that find applications in a variety of different contexts. Relaying on the error probability as cost function to evaluate their efficiencies [Hel76b], we schematize these procedures in terms of the following Minimum Error Discrimination (MED), and Minimum Error Classification (MEC) problems:

MED problem:– *Given a set of known quantum states* $\{\hat{\rho}_k\}_{k=1}^K$ *and probabilities* $\{p_k\}_{k=1}^K, \sum_k p_k = 1$ *and an unknown quantum state* $\hat{\rho} \in \{\hat{\rho}_k\}$ *drawn from the set with probability* $p \in \{p_k\}$*, find the POVM measurement operators* $\{\hat{\Pi}_k\}_{k=1}^K$ *that allows to identify* ρ *with minimum probability of error* $P_e^{(\mathrm{MED})}$*, or equivalently, with maximum probability of correct decision* $P_c^{(\mathrm{MED})} = 1 - P_e^{(\mathrm{MED})}$*,*

$$P_c^{(\mathrm{MED})} = \sum_{k=1}^{K} p_k \,\mathrm{tr}\left[\hat{\Pi}_k \hat{\rho}_k\right] . \tag{4.1}$$

It is worth reminding that for the special case with $K = 2$ an explicit solution for the MED problem is provided by the Helstrom theorem according to which the maximum value of $P_c^{(\mathrm{MED})}$ is achieved by a binary projective measurement associated with positive and negative part of the operator $p_1\hat{\rho}_1 - p_2\hat{\rho}_2$, leading to the optimal expression

$$P_{c,\max}^{(\mathrm{MED})} = \frac{1}{2}\left(1 + \mathrm{tr}|p_1\hat{\rho}_1 - p_2\hat{\rho}_2|\right) , \tag{4.2}$$

which we shall employ in the following as a benchmark for the efficiency of our schemes.

MEC problem:– *Given a training set of quantum states* $\{\hat{\rho}_k\}_{k=1}^n$ *and a set of labels* $\{y(k) \mid y(k) \in [1,\ldots,K]\}_{k=1}^n$ *that associate each sample to its class* $y(k)$*, and given an unknown testing set of quantum states* $\{\hat{\rho}_r\}_{r=1}^s$ *and labels* $\{z(r)|\ z(r) \in [1,\ldots,K]\}_{r=1}^s$*, find the POVM measurement operators* $\{\hat{\Pi}_z\}_{z=1}^K$ *that identify their classes with minimum error probability* $P_e^{(\mathrm{MEC})}$*, or equivalently, with maximum probability of correct decision* $P_c^{(\mathrm{MEC})} = 1 - P_e^{(\mathrm{MEC})}$*,*

$$P_c^{(\mathrm{MEC})} = \frac{1}{s}\sum_{r=1}^{s} \mathrm{tr}\left[\hat{\Pi}_{z(r)}\hat{\rho}_r\right] . \tag{4.3}$$

Notice that at variance with the MED problem, for the MEC problem a (complete classical) description of the quantum states of the training set

$\{\hat{\rho}_k\}_{k=1}^n$ is not known, nor it is the a priori probability of each class which is often inferred from the relative amount of the labels. Notice also that, even though in the subsequent sections of our manuscript we shall not adopt such options, in the general MEC setting i) the samples of the training set are not necessarily organized into cluster of identical copies, and ii) the testing samples $\{\hat{\rho}_r\}_{r=1}^s$ are not included in the training set $\{\hat{\rho}_k\}_{k=1}^n$.

4.2.2 Quantum optical setting

In the rest of the paper we shall consider the case where the system of interest is a single optimal mode of the electro-magnetic field described by the annihilation and creation operators $\hat{a}$, $\hat{a}^\dagger$ fulfilling canonical commutation rules, $[\hat{a}, \hat{a}^\dagger] = 1$ [Ser17, WM08]. In this context we focus on a MEC problem where the training set is formed by identical copies of $K = 2$ coherent states

$$|\alpha_k\rangle = \hat{D}(\alpha_k)|\emptyset\rangle \ , \qquad k \in \{1, 2\} \ , \tag{4.4}$$

whose complex amplitudes α_1 and α_2 are unknown (in this expression $|\emptyset\rangle$ and $\hat{D}(\alpha) = \exp[\alpha\hat{a}^\dagger - \alpha^*\hat{a}]$ are respectively the vacuum state and the displacement operator of the model). Specifically we shall work under the assumption of having n copies of each training state, and that the density matrix we need to classify is guaranteed to be a coherent state $|\delta\rangle$ that coincides either with $|\alpha_1\rangle$ or with $|\alpha_2\rangle$, with flat prior probabilities (i.e. $p_1 = p_2 = 1/2$). The resulting global input state we operate can hence be expressed in the following multi-mode compact form

$$\left|\alpha_1^{\otimes n}, \alpha_2^{\otimes n}, \delta\right\rangle = |\alpha_1\rangle^{\otimes n} \otimes |\alpha_2\rangle^{\otimes n} \otimes |\delta\rangle \ , \tag{4.5}$$

which in principle is characterized by 4 unknown real parameters (the complex numbers α_1 and α_2), and by one quantum binary variable (the testing state $|\delta\rangle \in \{|\alpha_1\rangle, |\alpha_2\rangle\}$). In the MED version of the problem, i.e. when the complex quantities α_1, α_2 are assigned or equivalently when the number n of the training copies of the MEC problem are infinitely many so that the values of the amplitudes can be recovered through quantum process tomography, the optimal success probability can be computed as in Eq. (4.2) leading to the value

$$\begin{aligned} P_{c,\max}^{(\mathrm{MED})} &= \frac{1}{2}\left(1 + \sqrt{1 - 4p_1p_2|\langle\alpha_1|\alpha_2\rangle|^2}\right) \\ &= \frac{1}{2}\left(1 + \sqrt{1 - 4p_1p_2\ \mathrm{e}^{-|\alpha_1-\alpha_2|^2}}\right) \ , \end{aligned} \tag{4.6}$$

which can be attained via the Dolinar detection scheme which we review in the next section. The procedure we have in mind to solve the MEC problem

for finite n is a variation of the such scheme that relays on basic linear optical manipulations of the state $\left|\alpha_1^{\otimes n}, \alpha_2^{\otimes n}, \delta\right\rangle$ to compensate for the absence of classical information on the values of the amplitudes α_1 and α_2. We call such procedure agnostic-Dolinar receiver and we present it in Section 4.3.

4.2.3 Dolinar receiver

As anticipated the Dolinar receiver is an experimental technique that allows one to practically attain the optimal threshold limit (4.6) for a binary MED problem aimed to discriminate between two assigned coherent input states $|\alpha_1\rangle$ and $|\alpha_2\rangle$ which are produced with prior probabilities p_1 and p_2. It is worth noticing that in this special context, due to the fact that the values of the complex amplitudes α_1 and α_2 are known, one can always reduce the problem to case of a symmetric configuration in which $|\alpha_1\rangle$, $|\alpha_2\rangle$ are traded with the couple $|\pm\bar{\alpha}\rangle$ with $\bar{\alpha} = (\alpha_1 - \alpha_2)/2$, or equivalently to a maximally energetically unbalanced setting where instead $|\alpha_1\rangle$ and $|\alpha_2\rangle$ get replaced by $|2\bar{\alpha}\rangle$ and $|\emptyset\rangle$, respectively. Such mappings in fact simply relay on acting on the input $|\delta\rangle$ via optical displacements, i.e. transformations that can be physically implemented by mixing the signal with an intense coherent ancillary state through a beam-splitter of high transmissivity. Specifically in the cases we are considering this accounts in replacing $|\delta\rangle$ with

$$\hat{D}\left(-\beta\right)|\delta\rangle = e^{(\alpha\beta^* - \alpha^*\beta)/2}\,|\delta - \beta\rangle \ , \tag{4.7}$$

with $\beta = (\alpha_1 + \alpha_2)/2$ for the symmetric configuration, and $\beta = -\alpha_2$ for the maximally energetically unbalanced setting. In view of these facts in the following paragraphs, without loss of generality we shall assume the symmetric setting posing $\alpha_1 = -\alpha_2 = \alpha$. We also point out that in our presentation of the Dolinar scheme we shall rely to the continuous-time formulation of the problem discussed in [ADPP11a] – see however Appendix A.1 for a description based on sequence of beam splitters and photon–detectors.

The Dolinar receiver works by continuously applying a displacement $\hat{D}(\gamma_k(t))$ on the input state and performing a photon counting on the displaced signal (see upper panel of Fig. 4.1). The displacement is optimized such that the parity of the counting at the end of the signal gives the final estimate of the input coherent state. The rationale of the scheme follows the same idea of another suboptimal scheme, the Kennedy receiver [Ken73]. The key idea behind the Kennedy receiver is to rigidly shift the two states by α to obtain the pair $\{|2\alpha\rangle, |0\rangle\}$ (i.e. to map the symmetric setting into the maximally energetically unbalanced one), and then perform photon counting.

The vacuum always counts zero photons, and thus the Kennedy receiver uses $\hat{\Pi}_+ = \mathbb{1} - |\emptyset\rangle\langle\emptyset|$, $\hat{\Pi}_- = |\emptyset\rangle\langle\emptyset|$ as the POVM for the discrimination. This apparatus does not reach the optimal Helstrom bound (4.6), and this gap is filled exactly by the Dolinar receiver [Dol73a, ADPP11a, TSvLL05a], where the optimal shift is defined from an optimization procedure. More precisely, consider the input field for our system $\psi_k(t)$, $0 < t < T$, corresponding to the coherent state $|\pm\alpha\rangle$, being represented by

$$\psi_k(t) = \pm\psi e^{i\omega_0 t}, \qquad (k = \pm)\ , \tag{4.8}$$

with ω_0 the optical pulsation frequency and T the pulse duration, that are linked through the normalization condition

$$|\alpha|^2 = \int_0^T |\psi(t)|^2\, \mathrm{d}t = |\psi_k|^2\, T. \tag{4.9}$$

It can be shown that all the following results do not depend on our choice of T and thus, for clarity reasons, we set $T = 1$ and $\psi = \alpha$. This input signal is displaced by a value $\gamma_k(t)$, which is classically controlled via feedback, and the resulting sum signal is monitored with a photon counter. Every time our counter "clicks", the added signal is discontinuously changed from $\gamma_+(t)$ to $\gamma_-(t)$ and vice-versa, which are the displacement employed when the provisional estimation for the quantum state is $z(t) = +$ or $z(t) = -$ corresponding to the sign of $|\pm\alpha\rangle$. Equivalently, the parity of the total number of photons counted until the current time t gives the estimation for the quantum state. The final discrimination result is declared at the end of the signal, $t = T = 1$.

We can use the following argument to find the optimal choice for $\gamma_k(t)$. The provisional correct decision probability at time t can be written as

$$P_c(t) = P[z(t) = +|k = +]p_+ + P[z(t) = -|k = -]p_-. \tag{4.10}$$

Let us now assume that $k = +$, that is, the actual coherent to discriminate is represented by the input field $\psi_+(t)$ of Eq. (4.8). Then, the process $z(t)$ can be interpreted as a telegraph process driven by non-homogeneous Poisson processes [ADPP11a] with rates

$$\begin{aligned} \lambda(t) &= |\psi_+(t) - \gamma_+(t)|^2\,, \\ \mu(t) &= |\psi_+(t) - \gamma_-(t)|^2\,, \end{aligned} \tag{4.11}$$

that allows the evaluation of the differential equations for the conditional probabilities of correct decision $q_+(t) = P[z(t) = +|k = +]$ and $q_-(t) =$

$P[z(t) = -|k = -]$ as

$$\begin{aligned}\frac{\mathrm{d}q_+(t)}{\mathrm{d}t} &= \mu(t) - [\lambda(t) + \mu(t)]q_+(t),\\ \frac{\mathrm{d}q_-(t)}{\mathrm{d}t} &= \mu(t) - [\lambda(t) + \mu(t)]q_-(t).\end{aligned} \tag{4.12}$$

Hence, the differential equation for the correct detection probability results

$$\frac{\mathrm{d}P_c(t)}{\mathrm{d}t} = \frac{q'_+ + q'_-}{2} = \mu(t) - [\lambda(t) + \mu(t)]P_c(t)\ . \tag{4.13}$$

We can now extremize with respect to $\gamma_+(t) = -\gamma_-(t)$ at each fixed time to find the optimal displacement, which leads to the differential equation

$$\begin{aligned}\frac{\mathrm{d}P_c(t)}{\mathrm{d}t} &= -4|\alpha|^2\frac{P_c(t)(1-P_c(t))}{1-2P_c(t)}\\ &= |\alpha|^2\left(1 - 2P_c(t) - \frac{1}{1-2P_c(t)}\right)\end{aligned} \tag{4.14}$$

with solution

$$P_{c,\mathrm{Dol}}^{(\mathrm{MED})}(t) = \frac{1}{2}\left(1 + \sqrt{1 - 4p_+p_-\mathrm{e}^{-4|\alpha|^2 t}}\right)\ , \tag{4.15}$$

which for $t = 1$ reaches the maximum value (4.6) dictated by the Helstrom bound [Hel76b]. An experimental realization of this apparatus has been realized in [CMG07a].

4.3 Building an agnostic-Dolinar Receiver

In this section we present the novel results of our work, which lead to a scheme for solving the MEC problem associated with a binary set of coherent inputs introduced in Section 4.2.2. As a preliminary step, following the discussion at beginning of Section 4.2.3, in Section 4.3.1 we show that via some trivial physical manipulations of the input data we can always restrict the analysis to the special case of a binary MEC problem where $\alpha_2 = -\alpha_1$, hence reducing from 4 to 2 the number of unknown real parameters associated with the input state Eq. (4.5). In Section 4.3.2 we find the equivalent of the Helstrom bound in Section 4.2.2 for the MEC scenario: as we shall see the attainability of such optimal threshold relay on the possibility of implementing highly non-linear optical processes which represent an impressive challenge for current technology. In Section 4.3.3 we hence focus on a more realistic

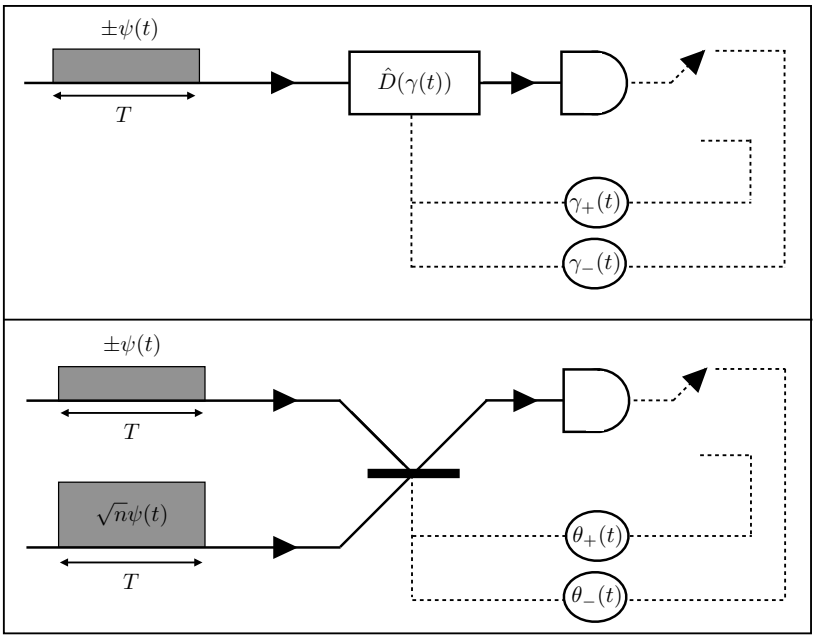


Figure 4.1: Time continuous description of the conventional Dolinar receiver (upper panel) and the agnostic-Dolinar receiver (lower panel): continuous black lines represent optical signals, dashed lines represent instead classical control lines. In the upper panel, the rectangle with the $\hat{D}(\gamma(t))$ symbol represents a time-dependent displacement gate, which is followed by a photon counter that switch between two classically controlled quantities represented in the circles. In the lower panel, the crossing with an extra horizontal rectangle is a beam splitter with time-dependent reflectivity $\theta_k(t) \in \{\theta_+(t), \theta_-(t)\}$, which are classically controlled and selected by the photon counter.

procedure based on an adaptive scheme where the displacements $\hat{D}(\gamma_k(t))$ of the original Dolinar receiver are replaced by partial coherent mixing with a fraction of the copies of the training set. In this context we show that if the value $|\alpha_1 - \alpha_2|$ is known (an assumption which would be trivially granted in the MED version of the problem but not in the MEC scenario where it allows to reduce the number of the unknown real parameters needed from 2 to 1), the new setting attains a probability of success that already for medium size values of n, approaches the one of the non-linear optimal bound of Section 4.3.2. In Section 4.3.4 we fix the issue associated with the lack of knowledge of the parameter $|\alpha_1 - \alpha_2|$ by exploiting part of the copies of the training set to obtain a preliminary estimation of such term: the performance of the resulting scheme is hence studied and compared with those one would obtain by using a mis-calibrated Dolinar scheme.

4.3.1 Mapping the MEC problem into a symmetric scenario

Aim of this section is to show that when studying the MEC problem introduced in Section 4.2.2, we can safely assume the amplitudes of the unknown coherent states of the training set, to have opposite phases and equal absolute values (i.e. $\alpha_2 = -\alpha_1$). This simplification is analogous to the reduction of the general MED problem to a symmetric configuration: in the present case however this formal passage is slightly more subtle due to the fact we do not have prior classical knowledge of α_1 and α_2.

A key ingredient of the analysis is represented by what we may call a m-modes *concentrator gate* $\hat{U}_C^{(m)}$ [SGA15a, SZBH09, Guh11, SZBH08], i.e. a m-modes unitary transformation implementable via array of properly concatenated beam-splitters, that acting on a collection of m copies of a generic (possibly unknown) coherent state $|\alpha\rangle$ manages to move all their photons in a single output mode via the mapping

$$|\alpha\rangle^{\otimes m} \longmapsto \hat{U}_C^{(m)}|\alpha\rangle^{\otimes m} = |\sqrt{m}\alpha\rangle \otimes |\emptyset\rangle^{\otimes m-1} \ . \tag{4.16}$$

Applying this to the $(2n+1)$-modes input state (4.5) that formally defines the MEC problem we are facing, we can map it into an equivalent form where all photons are concentrated into the following 3-modes coherent state

$$\left|\sqrt{n}\alpha_1, \sqrt{n}\alpha_2, \delta\right\rangle = \left|\sqrt{n}\alpha_1\right\rangle \otimes \left|\sqrt{n}\alpha_2\right\rangle \otimes |\delta\rangle \ , \tag{4.17}$$

(the net operation involves a collection of extra $2(n-1)$ irrelevant vacuum states $|\emptyset\rangle$). Notice also that with a 3–port beam splitter [Ser17] defined by the 3×3 scattering matrix

$$S_n = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{4n+2}} & \frac{1}{\sqrt{4n+2}} & -\sqrt{\frac{2n}{2n+1}} \\ \sqrt{\frac{n}{2n+1}} & \sqrt{\frac{n}{2n+1}} & \sqrt{\frac{1}{2n+1}} \end{bmatrix} \ , \tag{4.18}$$

we can then unitarily transform (4.17) (and hence (4.5)) into the further equivalent form

$$\left|\sqrt{2n+1}\alpha'\right\rangle \otimes |\delta'\rangle \otimes \left|\tfrac{\delta+n(\alpha_1+\alpha_2)}{\sqrt{2n+1}}\right\rangle \ , \tag{4.19}$$

where now

$$\alpha' = \sqrt{\tfrac{2n}{2n+1}}\,\tfrac{\alpha_1-\alpha_2}{2} \ , \tag{4.20}$$

and where

$$\delta' = \sqrt{\tfrac{2n}{2n+1}}\left(\tfrac{\alpha_1+\alpha_2}{2} - \delta\right) \ , \tag{4.21}$$

is a variable that for $\delta = \alpha_1, \alpha_2$ assumes the values $\pm\alpha'$. Therefore, since the coherent state $\left|\sqrt{2n+1}\alpha'\right\rangle$ can be mapped into $|\alpha'\rangle^{\otimes(2n+1)}$ via the action of the inverse of a concentrator gate $\hat{U}_C^{(2n+1)}$ our original MEC problem associated with the input (4.5), can be casted into the new MEC problem where starting from a collection of $2n+1$ copies of the coherent state $|\alpha'\rangle$ we are asked to decide wether the state $|\delta'\rangle$ is either equal to $|\alpha'\rangle$ or to $|-\alpha'\rangle$. Notice that in doing so we are implicitly neglecting the last coherent state component of Eq. (4.19): this however does not represent a huge loss of information since the residual dependence that such term bares upon δ vanishes in the asymptotic limit of $n \to \infty$, and in any case the analysis shows that a scheme that is capable to efficiently solve a MEC in the symmetric scenario can also be applied to the generic one. Notice further that via the action of a phase shifter gate aimed to flip the sign of the amplitude of an incoming input state, we can also convert the $2n+1$ copies of $|\alpha'\rangle$ into a state of the form $|\alpha'\rangle^{\otimes m} \otimes |\alpha'\rangle^{\otimes 2n+1-m}$ with $0 \leq m \leq 2n+1$. Relying on these observations in the remainder of the paper we will thus focus on the symmetric version of our MEC problem where starting from the beginning it is assumed $\alpha_1 = -\alpha_2 = \alpha$, hence replacing the input state (4.5) with the vector

$$\left|\alpha^{\otimes n}, \delta\right\rangle = |\alpha\rangle^{\otimes n} \otimes |\delta\rangle \ , \tag{4.22}$$

characterized by 2 unknown real parameters (the phase and the absolute values of the complex number α), and by the quantum binary variable $|\delta\rangle \in \{|\pm\alpha\rangle\}$ [NB. formally speaking in the above expression the total number of copies of $|\alpha\rangle$ we can extract from (4.5) requiring $\alpha_1 = -\alpha_2$ would be $2n$: hereafter however we shall reparametrize this with n just to allow for the possibility of having an odd number of input copies].

4.3.2 Optimal bound for the problem

As already anticipated in the asymptotic limit $n \gg 1$, the optimal upper bound for the success probability of a generic apparatus aimed to solve the MEC problem associated with the input (4.22) reduces to the Helstrom limit (4.6) attainable via the Dolinar scheme, i.e. the quantity

$$P_{c,\max}^{(\mathrm{MED})} \ = \ \frac{1}{2}\left(1 + \sqrt{1 - \mathrm{e}^{-4|\alpha|^2}}\right) \ . \tag{4.23}$$

Estimating the optimal MED performance in the finite-copy case can be useful to compare our results with a fundamental bound. In this section we will find this bound under the assumption that the protocol we use is phase invariant, i.e. insensitive to the phase value of the amplitude α that

enters in Eq. (4.22), a constraint which is reasonable to impose in the MED scenario where no prior info on α is granted. For this purpose, first of all we invoke once more the action of a concentrator gate (4.16), to replace $|\alpha^{\otimes n}, \delta\rangle$ with a two mode input state $|\sqrt{n}\alpha, \delta\rangle$. Then we focus on two-element POVM $\{\hat{E}_+, \hat{E}_- = \hat{I} - \hat{E}_+\}$ which, acting globally on the two modes of the model, aims to discriminate the density matrix $\hat{\rho}_+ = |\sqrt{n}\alpha, \alpha\rangle \langle\sqrt{n}\alpha, \alpha|$ from $\hat{\rho}_- = |\sqrt{n}\alpha, -\alpha\rangle \langle\sqrt{n}\alpha, -\alpha|$ under phase invariant assumptions. It is worth stressing that a similar calculation was performed in Ref. [SGA15a] for a slightly different setting where the two states under scrutiny were $|\sqrt{n}\alpha, \alpha\rangle$ and $|\sqrt{n}\alpha, 0\rangle$ and where the analysis was confined in large n limit: as we shall see in the following, at variance with those results, due to the symmetric structure of the inputs we employ, our analysis allows us to present closed analytical expressions also for the finite n limit. Specifically we associate $\hat{E}_+$ to $\hat{\rho}_+$ and $\hat{E}_-$ to $\hat{\rho}_-$, and we enforce the phase invariant constraint by requiring them to commute with the global phase operator $e^{i\phi(\hat{n}_1+\hat{n}_2)}$ with $\hat{n}_1 = \hat{a}_1^\dagger \hat{a}_1$ and $\hat{n}_2 = \hat{a}_2^\dagger \hat{a}_2$ being the number operators of the two modes of the model. By Schur's lemma [HH03] it then follows that the POVM elements must satisfy the identities $\hat{E}_\pm = \sum_{m=0}^\infty \hat{\Pi}_m \hat{E}_\pm \hat{\Pi}_m = \sum_{m=0}^\infty \hat{E}_{+,m}$, where $\hat{\Pi}_m$ is the projector on the subspace total photon number $n_1 + n_2 = m$ and $\{\hat{E}_{+,m}, \hat{E}_{-,m} = \hat{I}_m - \hat{E}_{+,m}\}$ is a binary POVM in this subspace. For any two states $\hat{\rho}_+$ and $\hat{\rho}_-$, POVMs $\{\hat{E}_\pm\}$ with this property, the probability of error satisfies

$$\begin{aligned} P_e &= \frac{1}{2}\left(1 - \sum_{m=0}^{\infty} \frac{\mathrm{tr}[\hat{E}_- \hat{\Pi}_m \hat{\rho}_+ \hat{\Pi}_m + \hat{E}_+ \hat{\Pi}_m \hat{\rho}_- \hat{\Pi}_m]}{2}\right) \\ &\geq \frac{1}{2}\left(1 - \sum_{m=0}^{\infty} \frac{\left\| \hat{\Pi}_m (\hat{\rho}_+ - \hat{\rho}_-) \hat{\Pi}_m \right\|_1}{2}\right) . \end{aligned} \tag{4.24}$$

where the inequality comes from the Helstrom bound. Remembering that in our case $\hat{\rho}_\pm$ are the coherent states $|\sqrt{n}\alpha\rangle\langle\sqrt{n}\alpha| \otimes |\pm\alpha\rangle\langle\pm\alpha|$ it follows that the optimal choice for $\hat{E}_\pm$ is given by

$$\hat{E}_\pm := \sum_{m=0}^{\infty} |m; \pm\rangle\langle m; \pm| , \tag{4.25}$$

with

$$|m, \pm\rangle := \sum_{n_1+n_2=m} \sqrt{\binom{m}{n_1}} \frac{\sqrt{n}^{n_1} (\pm 1)^{n_2}}{\sqrt{n+1}^m} |n_1, n_2\rangle . \tag{4.26}$$

Accordingly as shown in Appendix A.2, the minimum error probability (Eq. (4.24)) reduces to

$$P_{e,\min}^{(\mathrm{MEC},n)} = \frac{1}{2}\left(1 - \frac{1}{2}\sum_{m=0}^{\infty}\mathfrak{p}(m;\mu)\sqrt{1-\left(\tfrac{N-1}{N+1}\right)^{2m}}\right) , \tag{4.27}$$

where $\mu = \sqrt{n+1}\ \alpha$ and $\mathfrak{p}(m;\mu)$ is the probability of drawing m from a poissonian of mean $|\mu|^2$, namely

$$\mathfrak{p}(m;\mu) = \frac{|\mu|^{2m}}{m!}\exp[-|\mu|^2]. \tag{4.28}$$

Notice in particular that, for fixed $|\alpha|^2$, the above expression admits the following asymptotic expansion at large n

$$\begin{aligned} P_{e,\min}^{(\mathrm{MEC},n)} \simeq & \frac{1}{2}\left[1 - \frac{1}{2}\left(\sqrt{1-e^{-4|\alpha|^2}} - \frac{1}{n}\frac{2|\alpha|^2 e^{-4|\alpha|^2}}{\left(1-e^{-4|\alpha|^2}\right)^{3/2}}\right)\right] \\ & + O\left(\tfrac{1}{n^2}\right), \end{aligned} \tag{4.29}$$

with the leading term corresponding to value dictated by the Helstrom limit (4.23). The same analysis can be repeated for input states of our problem with a given prior $p(\alpha)$ by simply replacing $\hat{\rho}_{\pm}$ with

$$\hat{\rho}_{\pm}^{(\mathrm{ave})} = \int_{\mathbb{C}^2} \mathrm{d}\alpha\ p(\alpha)\left|\sqrt{n}\alpha\right\rangle\!\left\langle\sqrt{n}\alpha\right| \otimes |\pm\alpha\rangle\langle\pm\alpha| \ , \tag{4.30}$$

obtaining in this case the following optimal minimum error probability

$$\bar{P}_{e,\min}^{(\mathrm{MEC},n)} = \frac{1}{2}\left(1 - \frac{1}{2}\sum_{m=0}^{\infty}\bar{\mathfrak{p}}(m)\sqrt{1-\left(\frac{n-1}{n+1}\right)^{2m}}\right) , \tag{4.31}$$

where now

$$\bar{\mathfrak{p}}(m) = \int_{\mathbb{C}^2} \mathrm{d}\alpha\ p(\alpha)\ \mathfrak{p}(m;\sqrt{n+1}\alpha) \ , \tag{4.32}$$

is the average photon number distribution.

4.3.3 Agnostic-Dolinar receiver with prior information on the input mean photon number

The implementation of the optimal covariant measure (4.25) is highly non trivial as it requires to discriminate between non-orthogonal states $|m,\pm\rangle$

that involve complex superposition of two-mode Fock states (see Eq. (4.26)). To compensate for this here we introduce a preliminary version of the agnostic-Dolinar scheme that assumes that only the phase of the complex parameter α of Eq. (4.22) to be unknown, but grants full knowledge about the mean photon number $|\alpha|^2$ of the inputs. In other words we interpolate between the symmetric version of the MEC problem defined at the end of the previous section (2 unknown real terms and one quantum binary variable), and the corresponding MED problem (zero unknown real terms and one quantum binary variable). As schematically shown in the lower panel of Fig. 4.1, the basic idea is to replace the displacement operations of the original Dolinar configuration whose values γ_k assume full knowledge of α, with coherent mixing of the testing input with the single-mode concentrated version $|\sqrt{n}\alpha\rangle$ of the training copies, via a beam-splitter operation characterized by a time–dependent reflectivity $\theta_k(t)$, where k can be either $+$ or $-$. Because of the symmetry of the problem, we will call $\theta(t) = \theta_+(t) = \theta_-(t) + \pi$. Note that in the alternative formulation of the Dolinar receiver [TSvLL05a] discussed in Appendix A.1 where the input coherent state is sliced by the sequence of beam splitter and on each slice a displacement and photon counting process is applied, the scheme of the current section substitute the displacement operations with additional beam splitters, which mix slices of the n training copies and the sliced input field in an optimized way to give output rates Eq. (4.33). It is finally worth stressing that, since we not relay on reference signals, by construction, the proposed detection strategy is explicitly phase insensitive: accordingly the optimal bound (4.27) constitute a proper reference for testing the efficiency of the scheme.

In order to evaluate the optimal $\theta(t)$ and maximize the correct decision probability we follow the same procedure of Section 4.2.3, that we will now discuss with a more detail. First of all, following the same procedure, we define our new $\lambda(t), \mu(t)$ as

$$\begin{aligned} \lambda(t) &= |\alpha|^2 \left|\cos\theta(t) - \sqrt{n}\sin\theta(t)\right|^2, \\ \mu(t) &= |\alpha|^2 \left|\cos\theta(t) + \sqrt{n}\sin\theta(t)\right|^2. \end{aligned} \tag{4.33}$$

Let us now assume that $k = +$. The number of photons counted in an interval $(t, t + \Delta t]$ is a Poisson variable $N(t, t + \Delta t)$ with parameter $\lambda(t)\Delta t$ or $\mu(t)\Delta t$ depending whether the provisional hypothesis is respectively $z(t) = +$ or $z(t) = -$. These rates allows us to evaluate the conditional probabilities of correct decision $q_+(t) = P[z(t) = +|k = +]$ and $q_-(t) = P[z(t) = -|k = -]$ following the Dolinar receiver strategy, which changes the provisional hypothesis when a photon is detected [ADPP11a]. From the

difference equation

$$\begin{aligned}
&q_+(t+\Delta t)\\
&\quad = P[N(t,t+\Delta t)=0, z(t)=+|k=+]+\\
&\quad P[N(t,t+\Delta t)=1, z(t)=-|k=+]+o(\Delta t)\\
&\quad = P[N(t,t+\Delta t)=0|z(t)=+,k=+]q_+(t)+\\
&\quad P[N(t,t+\Delta t)=1|z(t)=-,k=+](1-q_+(t))+o(\Delta t)\\
&\quad = (1-\lambda(t)\Delta t)q_+(t)+\mu(t)\Delta t(1-q_+(t))+o(\Delta t),
\end{aligned} \tag{4.34}$$

follows the differential equation

$$\frac{\mathrm{d}q_+(t)}{\mathrm{d}t} = \mu(t) - [\lambda(t)+\mu(t)]q_+(t).$$

In a similar fashion, from the expression for $q_-(t+\Delta t)$ and employing symmetric arguments for the displacements, we get the other differential equation of Eq. (4.12). With this same procedure, it is obvious that the equation for $P_c(t)$ maintains the same form of Eq. (4.13), i.e.

$$\frac{\mathrm{d}P_c(t)}{\mathrm{d}t} = \mu(t) - [\lambda(t)+\mu(t)]P_c(t)\ , \tag{4.35}$$

where now however the terms $\mu(t)$ and $\lambda(t)$ are defined in Eq. (4.33) instead of Eq. (4.11) We can now extremize Eq. (4.35) to obtain an equation for the optimal control function $\theta^*(t) = \theta_{\mathrm{opt}}^{|\alpha|^2,(n)}(t)$ given $|\alpha|^2$ and n: this yields the solution

$$\tan(2\theta^*(t)) = \frac{\sqrt{n}}{n-1}\frac{1}{P_c^*(t)-\frac{1}{2}}, \tag{4.36}$$

where $P_c^*(t) = P_{c,\mathrm{opt}}^{|\alpha|^2,(n)}(t)$ is the the associated optimal probability of success that can be obtained by solving a differential equation which is more concisely expressed with the change of variable $\xi(t) = P_c^*(t) - 1/2$, i.e.

$$\begin{aligned}
\frac{\mathrm{d}\xi(t)}{\mathrm{d}t} &= |\alpha|^2\left(-\xi(t)(n+1)+\sqrt{(n-1)^2\xi^2(t)+n}\right)\\
&= |\alpha|^2\left[-2\xi(t)-(n-1)\left(\xi(t)-\sqrt{\frac{n^2}{(n-1)^2}+\xi(t)^2}\right)\right],
\end{aligned} \tag{4.37}$$

(notice the explicit dependence upon $|\alpha|^2$) which for $n \to \infty$ converges to Eq. (4.14). With the separation of variables we can formally integrate (4.37)

obtaining

$$\frac{1}{2}|\alpha|^2 t = -\frac{n-1}{4n}\tanh^{-1}\left(\frac{(n-1)\xi(t)}{\sqrt{(n-1)^2\xi^2(t)+n}}\right) + \frac{n+1}{8n}\Big[\tanh^{-1}\left(\frac{2\xi(t)(n+1)\sqrt{(n-1)^2\xi^2(t)+n}}{2(n^2+2)\xi^2(t)+n}\right) - \log\left(1-4\xi^2(t)\right)\Big]. \tag{4.38}$$

This expression cannot be inverted explicitly but it can be evaluated numerically. It turns out that the resulting $P_c^*(t)$ does not coincide with the optimal bound of Eq. (4.27). Still it remains close to such function being increasing in n and asymptotically reaching the performance of the Dolinar scheme given in Section 4.3.2. This is explicitly shown in Fig. 4.2, where we plot the resulting associated probability of error $P_e^* = 1 - P_c^*(t=1)$ as a function of the training set n for a known value of $|\alpha|^2$: notice that the asymptotic limit (4.23) is reached quickly for $n \approx 20$ even in the full quantum limit ($|\alpha| < 1$).

4.3.4 Agnostic-Dolinar receiver with no prior information on the input mean photon number

The scheme of the previous Section requires the exact knowledge of $|\alpha|^2$, which is not granted in the original MEC problem. Here, we compensate for such lack of information by splitting the n training copies of $|\alpha\rangle$ into two sets, one of size m used to obtain an estimate of the value of $|\alpha|^2$, and the other set of size $n-m$ copies to realize the apparatus described in the former section. Obviously, *a priori* there is no optimal choice for the size of the two sets, as there is a trade–off between a good parameter estimate and the performance of the apparatus. Studying the optimal way to split our sets will be the aim of this section. To estimate the classical value of $|\alpha|^2$ we examine two strategies, photon counting and heterodyne detection. In the former case, the outcomes of the measurement are discrete values $k \in \mathbb{N}$ associated with the count of photons of the state $|\alpha\rangle$ which get distributed according to the poissonian probability $\mathfrak{p}(k;\alpha)$ defined as in Eq. (4.28). When applied to a coherent state $|\sqrt{m}\alpha\rangle$, we can obtain an estimate $\frac{k}{m}$ for $|\alpha|^2$. Note that due to the discrete nature of the outcomes, such estimate comes in discrete steps. Heterodyne detection is obtained by mixing the coherent state $|\alpha\rangle$ with a strong local oscillator with higher optical frequency [Ser17], and then measuring both quadratures. The measurement outcomes are continuous and can be represented with a complex value $\beta \in \mathbb{C}$ obtained with probability

$$P[\beta;\alpha] = \frac{\mathrm{e}^{-|\beta-\alpha|^2}}{\pi}. \tag{4.39}$$

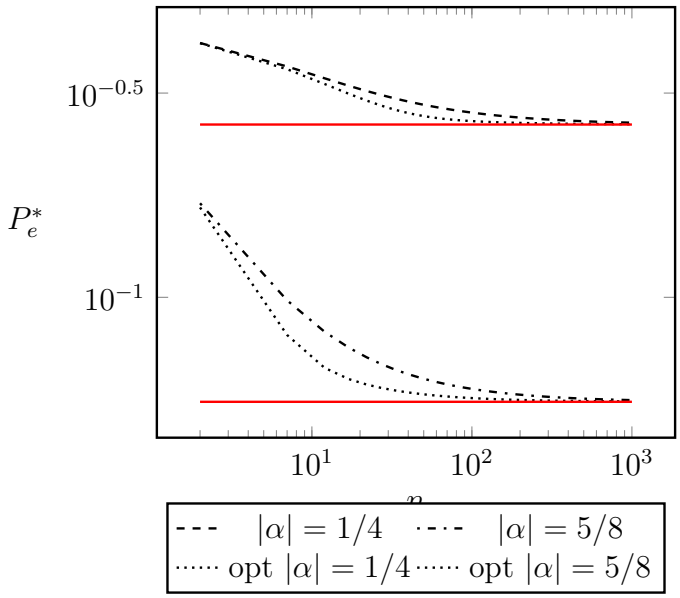


Figure 4.2: Error probability $P_e^* = 1 - P_c^*(t = 1)$ of an agnostic-Dolinar receiver (dashed and dash–dotted lines) as a function of the number n of training copies, for different (known) values of the mean photon number $|\alpha|^2$. The horizontal solid red lines correspond to the Helstrom values (4.23) attained by the conventional Dolinar scheme for the associated MED problem. The dotted lines correspond to the optimal error probability obtainable using the optimal bound for phase invariant scheme of Eq. (4.27).

The estimate for $|\alpha|^2$ can be obtained from the absolute value $|\beta|^2$ of the complex outcome, which is obtained with probability

$$P[|\beta|^2;\alpha] = \int_0^{2\pi} \frac{\mathrm{e}^{-(|\alpha|^2+|\beta|^2-2|\alpha\beta|\cos\phi)}}{\pi} |\beta|\,\mathrm{d}\phi$$
$$= 2\,|\beta|\,\mathrm{e}^{-|\alpha|^2-|\beta|^2} I_0(2\,|\alpha\beta|)\,, \tag{4.40}$$

with $I_0(\cdot)$ the modified Bessel function of the first kind.

To obtain the real performance of the apparatus, we take the expectation value over this probability distribution of the performance of the apparatus using the *wrong* estimate for $|\alpha|^2$. Namely, we use Eq. (4.36) and Eq. (4.38) with the amplitude $|\tilde{\alpha}|^2$ estimated from $|\sqrt{m}\alpha\rangle$ to obtain $\theta_{\mathrm{opt}}^{|\tilde{\alpha}|^2,(n-m)}(t)$. Averaging the performance for all the estimates gives the probability of correct decision as a function of α, n, m.

We compare the performance with the Helstrom bound (4.6), but also with an Estimate&Discriminate procedure which assumes of using all the n

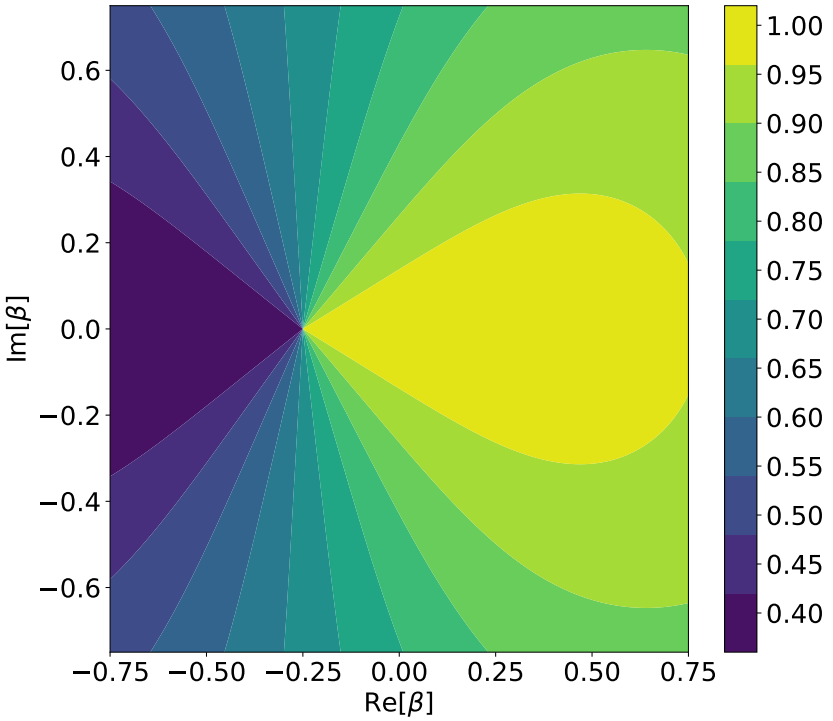


Figure 4.3: Density plot of the ratio between the probability of correct decision of a miscalibrated Dolinar receiver as in Eq. (4.41) and the optimal threshold $P_{\mathrm{c,max}}^{(\mathrm{MED})}$ of Section 4.3.2 as a function of the complex estimate β for $\alpha = 0.25$.

copies to obtain a classical estimate of α ($m = n$) and then apply the original Dolinar receiver. The performance of this straightforward method (which we dub miscalibrated Dolinar) is studied in the next section, Section 4.3.4, while the performances of the agnostic Dolinar are studied in Section 4.3.4 and Section 4.3.4.

Estimate&Discriminate scheme based on Miscalibrated Dolinar receiver

In the original setting of the Dolinar receiver, the value of α that uniquely determines the coherent state $|\alpha\rangle$ was known with arbitrary precision. In the Estimate&Discriminate approach we analyze here, the idea is to use all the n copies of $|\alpha\rangle$ of the state (4.22) to get an estimate β of α and then use this

to build up the Dolinar procedure.

To evaluate the performance of the scheme let us first consider what happens when a Dolinar receiver setup for the discrimination of $|\beta\rangle, |-\beta\rangle, \beta \neq \alpha$ is applied to the coherent states $|\alpha\rangle, |-\alpha\rangle$. For this purpose we can use Eq. (4.13), which is still valid, using the optimal displacement evaluated from β, obtaining a differential equation that can be solved analytically. As a result we get the following probability of success

$$P_{c,\mathrm{Dol}}^{(\beta;\alpha)} = \frac{1}{2} + \frac{\mathfrak{Re}\left[\alpha\beta^*\right]\left(1 - e^{-2\left(|\alpha|^2+|\beta|^2\right)}\right)}{\left(|\alpha|^2 + |\beta|^2\right)\sqrt{1 - e^{-4|\alpha|^2}}}, \tag{4.41}$$

which, by construction, is upper-bounded by the optimal value $P_{c,\max}^{(\mathrm{MED})}$ of Section 4.3.2) – see Fig. 4.3. The average success probability of the Estimate&Discriminate approach can now be obtained by averaging (4.41) with respect to probability $P[\beta;\alpha,n]$ of getting a certain value of β from our n copies of $|\alpha\rangle$, i.e.

$$P_{c,\mathrm{E\&D}}^{(\mathrm{MEC})} = \int_{\mathbb{C}^2} \mathrm{d}\beta \; P[\beta;\alpha,n] \; P_{c,\mathrm{Dol}}^{(\beta;\alpha)} \; . \tag{4.42}$$

A plot of this quantity as a function of n for few values of α can be found in Fig. 4.4 under the assumption that β is recovered via heterodyne detection, so that

$$P[\beta;\alpha,n] = \frac{n}{\pi} \exp\left[-n\,|\alpha-\beta|^2\right]. \tag{4.43}$$

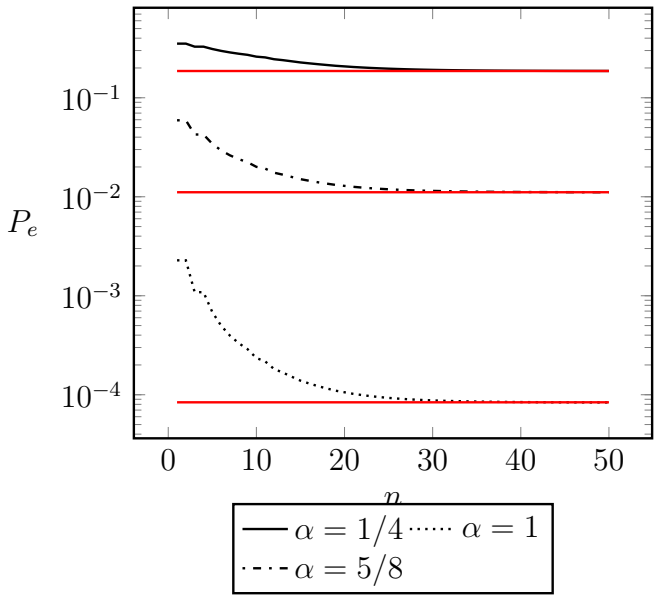


Figure 4.4: Error probability $1-P_{c,\mathrm{E\&D}}^{(\mathrm{MEC})}$ of the Estimate&Discriminate scheme base on a miscalibrated Dolinar receiver where we use the n training copies of $|\alpha\rangle$ of the input (4.22) to estimate the value of α via heterodyne mesurements. The red solid line is the Helstrom bound $1-P_{c,\max}^{(\mathrm{MED})}$ from (4.23).

Miscalibrated agnostic-Dolinar receiver

In this section, we study the performance of our classifier in the two-step procedure where we split our set of states into two different sets. The first one, of size m, is used to obtain an estimate of the value of $|\alpha|$, while the second one of size $n - m$ is used as input for the agnostic Dolinar receiver described in Section 4.3.3. The choice of the optimal m is highly non–trivial and we resort to a numerical procedure, here illustrated for $n = 15$. In the upper plot of Fig. 4.5 it is studied the dependence of the probability of correct decision on the size of the estimating set m with a photon counting estimator. Other values of n show a similar trend. The same setting, but with heterodyne-detection estimation, is plotted in the same Figure, in the lower panel. As can be seen from the figure, the optimal choice of m depends on the value of $|\alpha|$, but the optimal value belongs to a big plateau that allows us to ignore this dependence without losing too much performance. For this reason, we can choose *apriori* the value of m for each n, independent from α, looking at the plateau in the former figures. With this choice, we can finally compare our results with the Estimate&Discriminate strategy of Section 4.3.4 where all the training copies were used to estimate α with heterodyne detection and then a miscalibrated original Dolinar receiver was employed. These results are summarized in Fig. 4.6. The red solid line is the Helstrom bound, while the black lines are the Estimate&Discriminate performance for $n = 4$ and $n = 8$. The orange and blue lines correspond to photon counting and heterodyne detection respectively. We can clearly see a divergence in the optimal performance and the Estimate&Discriminate procedure, due to the difference in the concavity of the two plots. This does not happen with our strategy that remains close to the optimal bound. For low values of the distance between the states, the performance of the Estimate&Discriminate is slightly better than our, and this is due to our *apriori* choice of m, namely $(n = 4 \to m = 2, n = 8 \to m = 3)$, that is near the plateau for high (greater than 0.3) values of $|\alpha|$ but is not optimal for low values. If we chose the best m for each value of the distance, our performance would be better than the Estimate&Discriminate procedure, but this cannot be done for the reasons discussed above.

Performances in the presence of prior on $|\alpha|$

In this section we analyze the agnostic Dolinar scheme when we have a prior on the value of $|\alpha|$ but no information on the value of the phase $\arg \alpha$. In this case we can average the performance of Fig. 4.5 to obtain an expected probability of error for each size of the estimating set m, and choose the best

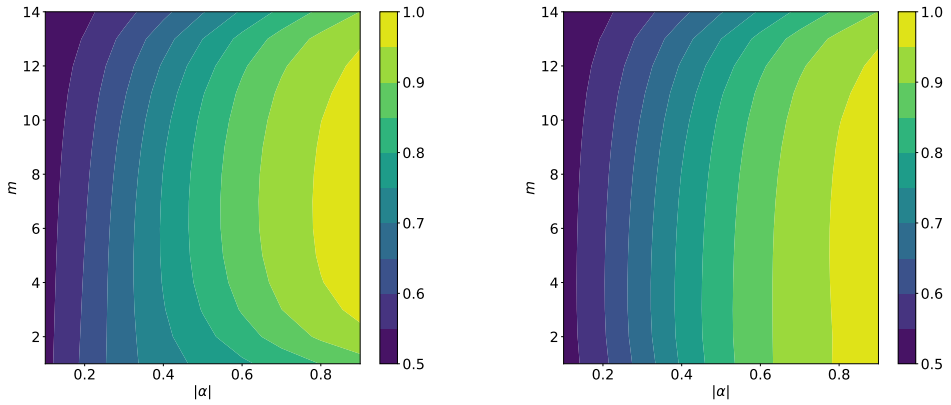


Figure 4.5: Probability of Correct decision computed as in indicated in Section 4.3.4 as a function α and the size m of the copies used to estimate it, for a total number of copies $n = 15$. The estimation of $|\alpha|$ is performed via photon counting in the upper plot and with heterodyne detection in the lower one.

m for the given prior. As an example we consider the case in which the prior distribution for $|\alpha|$ is given by the Rice distribution

$$\begin{aligned} p(|\alpha|\,;\sigma,x_c) &= \int_0^{2\pi} \frac{\exp\left[-\frac{(|\alpha|^2)+x_c^2-2x_c|\alpha|\cos\theta}{2\sigma^2}\right]}{2\pi\sigma^2}\,|\alpha|\,\mathrm{d}\theta \\ &= \frac{|\alpha|}{\sigma^2}\exp\left[-\frac{|\alpha|^2+x_c^2}{2\sigma^2}\right] I_0\left(\frac{|\alpha|\,x_c}{\sigma^2}\right), \end{aligned} \tag{4.44}$$

where I_0 is the modified Bessel function of the first kind. The obtained results are reported in Fig. 4.7, where the optimal error probability bound given in Eq. (4.31) is compared with the performance of our scheme, employing both heterodyne detection (blue lines) and photon counting (orange lines) to estimate $|\alpha|$, for $n = 4$ and $n = 8$. In this plot, which is evaluated with $\sigma = 0.1$ as a function of x_c, we can observe that with both measurements the performances remain close to the optimal ones as x_c increases, maintaining the ordering with respect to n (greater gives lower error probability).

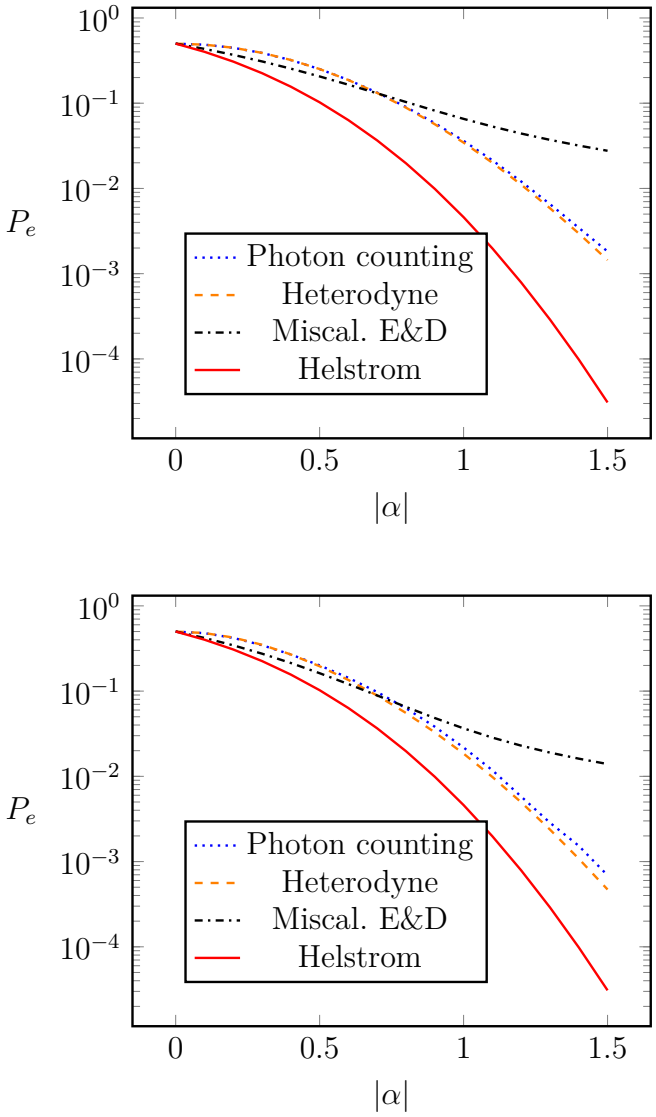


Figure 4.6: Error probability as a function of α for different classifiers, for size $n = 4$ (upper part) and $n = 8$ (lower part) of the training set. The red solid line is the Helstrom bound (4.23) while the agnostic-Dolinar receiver employing photon counting and heterodyne measurements are depicted with a blue dotted line and orange dashed line respectively. The miscalibrated E&D line is relative to the Estimate&Discriminate procedure based on full estimation with heterodyne detection and the use of a conventional (miscalibrated) Dolinar receiver.

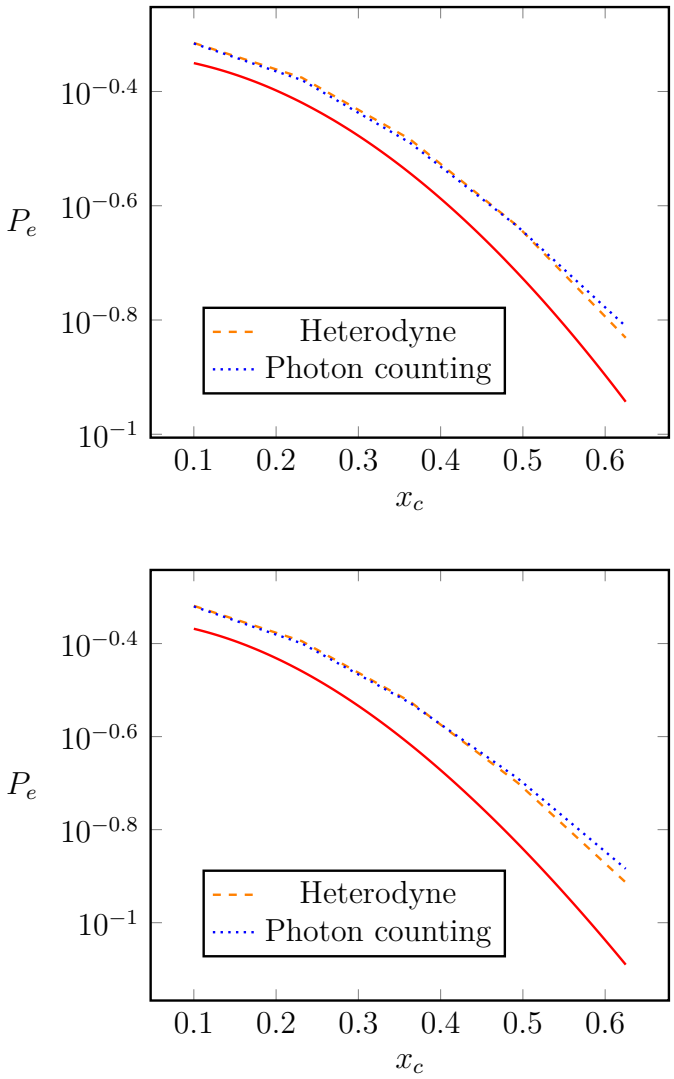


Figure 4.7: Error probability comparison given a prior Rice distribution Eq. (4.44) on $|\alpha|$ with $\sigma = 0.1$ as a function of x_c, for $n = 4$ (upper part) and $n = 8$ (lower part). The red solid lines represents the associated optimal bound for phase insensitive schemes in Eq. (4.31). The performance of the agnostic Dolinar receiver are plotted for heterodyne (dashed orange) and photon counting (blue dotted) measurements.

4.4 Conclusions

To summarize, here we have recalled the Minimum Error Discrimination and Minimum Error Classification problems of quantum states. The former, a central problem in Quantum Information Theory, assumes classical knowledge of the quantum states to discriminate. The latter, risen with the recent studies on machine learning, trades the classical description with the availability of multiple training copies assigned to the classes of quantum states to distinguish. In Quantum Optics, in the case of the binary discrimination of coherent states, an apparatus realizing the optimal discrimination is known (Dolinar receiver), while the corresponding for the classification problem is still missing. Between these two scenarios, we identify some intermediate setups with increasing level of classical knowledge: for instance, we can assume that the coherent states have opposite phases, or in addition to that, a prior distribution on the amplitude of the quantum states. We evaluate an optimal bound for this later problem, leveraging on the fact that the POVM associated with the optimal classifier must be phase invariant on the quantum states defined by the training copies and the state to distinguish. This bound asymptotically approaches the Helstrom limit in the limit of infinite training copies, that is the optimal bound for the discrimination problem.

We extend the Dolinar receiver with an agnostic formulation with and without prior information on the input mean photon number. In the case the prior is unknown, a fraction of the training copies is measured to estimate the mean photon number, either with a heterodyne measurement or via photon counting. The remaining training copies are employed in the classification device. We compare the performances of these schemes with the optimal bound previously evaluated and with a Miscalibrated Estimate&Discriminate apparatus, where all the training copies are employed in the amplitude estimation, used in a later stage by a Dolinar receiver. The trend of the schemes employing both heterodyne and photon counting measurements follows the optimal bound with a clear gap, but over–perform the Estimate&Discriminate strategy. This confirms and extends the results by [SGA15a], where the behaviour of the Estimate&Discriminate strategy was evaluated asymptotically in the number of training copies.

As future outlooks, one can narrow the gap with the optimal bound with adaptive strategies that estimate the coherent states amplitudes and perform a partial discrimination at the same time, and this is what is explained later, in Chapter 5. On the experimental side, the proposed classifiers can in principle be already implemented as they require state–of–the–art devices (beam splitters, phase shifters, photon counters and local laser sources) commonly present in current laboratories.

Chapter 5

Model-aware reinforcement learning for high-performance Bayesian experimental design in quantum metrology

5.1 Introduction

In recent times, the synergy between Machine Learning and quantum information has gained increasing attention. These two technological domains can be mutually beneficial in multiple ways. On one hand, quantum technologies, especially quantum computers, have the potential to address classic Machine Learning challenges, like classification and sampling, with both classical and quantum data [FHJ+20, BVM+21, BIS+22]. Conversely, traditional Machine Learning can augment quantum information tasks such as state preparation [BDS+18, ZWA+19, NBSN19, PEHM22], optimal quantum feedback [PPM23], error correction [FTWM18], device calibration [CGS+19, BED+21, NSP21, NPS21], characterization [NOL+21], and quantum tomography [PKB+20, QFN21, HNC+22]. This work fits in the latter category, using model-aware reinforcement learning [Mar21a, Mar21b, KLFM23, PPM23] (RL) to find optimized adaptive and non-adaptive control strategies for application-relevant tasks of quantum metrology and estimation [BZG24a]. The problem we are solving is that of optimal experimental design [Fis35], which has been already approached with ML techniques [Fos21, BCM+21, BBMO21, IFK+21, SM23, FIMR21]. It turns out that an estimation involves many non-differentiable steps, such as simulating the measurement and resampling from the posterior distribution. This could

potentially invalidate the application of model-aware RL. To address this issue, we propose an original combination of several techniques, including importance sampling, adding the log-likelihood of the sampled variables to the loss [PPM23], employing the reparametrization trick, and incorporating the Scibior and Wood correction [SW21]. Given a certain physical platform and metrological task, the set of tunable parameters in the experiment is identified. Then, an agent learns to optimally control them to minimize the error metric, through a gradient descent optimization procedure, based on the backpropagation of the derivatives through all the history of the estimation. The agent in question can be a small neural network, a decision tree, or a simple list of trainable controls that are sequentially applied. We abstracted this procedure, decoupled it from the particular sensor and physical platform, and packaged it in the qsensoropt library, available on PyPI, which can be used as a Swiss army knife for the optimization of quantum sensors. We demonstrate the broad applicability of our methodology by optimising a range of different examples on the nitrogen-vacancy (NV) center platform [CMZ+18, ROM+20], for single and multiparameter metrology, including both DC [FSB21] and AC magnetometry, decoherence estimation [ABH+22], and hyperfine coupling characterization [JSS+21]. For the photonic circuits, we studied multiphase discrimination, the agnostic Dolinar receiver [ZPFG21], and coherent states classification, both it the case the states are classically known and in the case in which they must be learnt from a quantum training set. In the domain of frequentist estimation, we studied the sensing of the detuning frequency in a driven optical cavity [FRTG22]. In this work only the applications to DC magnetometry and to the Dolinar receiver are presented, while the rest will be published in a future work [BZG24b]. Our findings indicate that model-aware RL outperforms traditional control strategies in multiple scenarios, beating also model-free RL. This work paves the way for researchers to speed up the search for optimal controls in quantum sensors, potentially hastening the advent of their broad industrial application.

The literature contains prior works addressing challenges similar to those addressed by our approach, which can be categorized into the following four classes. The first class encompasses the competitor approaches for optimization in quantum metrology using gradient descent. Meyer *et al.* proposed a variational toolbox for the optimization of measurements and states in multiparameter metrology [MBE21], but in contrast to our approach this doesn't allow Bayesian estimation nor it considers adaptive strategies. A similar tool is QuantEstimation [ZYY+22], which can't use neural networks as agents for the control. The two libraries QInfer [GFH+17] and Optbayesexpt [MBD21] can optimize the controls for a Bayesian experiment but only greedily, i.e. one measurement at a time, via an approximation of

the information gain per measurement. In [FSB21] Fiderer, Schuff, and Braun studies the application of model-free RL to the optimization of DC magnetometry. In [BLBSM23] a quantum comb-based approach to the simultaneous optimization of states and measurement for one-shot Bayesian experiments is put forward. The second class are those works that review the optimal control algorithms for quantum metrology, which are mainly based on the optimization of the Fisher information [LY17, XLL+19, ROM+20, SFB20, XWYW21, LZC+22, XFZ22, QZHL22]. These either lack coverage on Bayesian estimation or on the use of neural networks, or are applicable only to some specific platforms (like NV centers). The third class encompasses those theoretical works that advocate for the necessity of optimal control in quantum metrology and more or less conceptually shape the working principles of our approach, although without putting forward any implementation [BCM+21, BBMO21, KLFM23, VDPS23, GSG+23]. The fourth class contains the applications of variational quantum circuits to specific platforms and tasks. These are in general non-adaptive (with one exception [MGZ+21]) and can be Bayesian [KVS+21, MFP+22, KSVZ23] or based on the quantum Fisher information [KB23, HTS+23, YPC+22].

5.2 Theoretical overview of the framework

5.2.1 Encoding of the probe

Following the most common nomenclature in quantum metrology, we will define a quantum *probe* as a quantum system initialized in a reference state ρ. This probe is used to encode the d-dimensional vector of parameters $\boldsymbol{\theta} \in \Theta$ of interest, undergoing a controllable evolution, determined by the controls x, i.e. $\rho \to \rho_{x,\boldsymbol{\theta}} = \mathcal{E}_{x,\boldsymbol{\theta}}(\rho)$, where $\mathcal{E}_{x,\boldsymbol{\theta}}$ is a general LCPT map. A control is a tunable parameter that can be adjusted during the experiment, this could be for example the measurement duration, the detuning of the laser frequency driving a cavity, or a tunable phase in an interferometer. In a pictorial sense, the control parameters are all the buttons and knobs on the electronics of the experiment. A control can be continuous if it takes values in an interval, or discrete if it takes only a finite set of values (like on and off). The encoded parameters $\boldsymbol{\theta}$ can be a property of the environment, like a magnetic field acting on a spin in the NV center platform, or some degrees of freedom of the probe's initial state, like the parameter α of a coherent state of light $|\alpha\rangle$ in the agnostic Dolinar receiver. The same scheme can also be seen as a *communication protocol*, where Alice sends the state $\rho_{\boldsymbol{\theta}}$ to Bob, which has to decode the $\boldsymbol{\theta}$ vector. We will for the sake of generality call the quantum

system probe also in these cases. The idea is to perform a measurement on $\rho_{x,\boldsymbol{\theta}}$ to gain information about $\boldsymbol{\theta}$. An important distinction to be drawn here is that the term quantum parameter estimation refers in the literature to the situation in which we are given the encoded probe $\rho_{\boldsymbol{\theta}}$, in other words, we start from there and we can't act on the encoding, as opposed to quantum metrology where we are given access to the encoding process $\mathcal{E}_{x,\boldsymbol{\theta}}$ and not only to the final result. The implicit idea in parameter estimation [Hay05] is that the encoding has been carried out outside of the picture. Both in metrology and parameter estimation we assume that the encoding $\mathcal{E}_{x,\boldsymbol{\theta}}$ is applied many times or that we are given many copies of $\rho_{\boldsymbol{\theta}}$, so that we can collect some statistically relevant data by measuring all the copies, from which we infer the value of $\boldsymbol{\theta}$. Quantum metrology is a more general setting than parameter estimation and since the techniques developed here apply to quantum metrology they are also useful for parameter estimation. An example of parameter estimation would be receiving the radiation generated by a distribution of currents on a plane, which depends on the properties of the source, like the temperature for example [KB23]. In this scenario, the quantum probe is the radiation. Since the emission of the radiation happens by hypothesis in a far and inaccessible region, we don't have direct access to the quantum channel that performs the encoding, but only to the encoded states, which is the state of the radiated field at detection. An example of a quantum metrological task is the estimation of the environmental magnetic field with a spin, for which we can choose the initial state and the duration of the interaction. A parameter can be continuous or discrete. Naturally continuous parameters are the magnetic field and the temperature, for example. Some examples of discrete parameters are the sign of a signal and the type of the interaction between two quantum systems [GFK+21]. When discrete parameters are present, we are in the domain of values discrimination. In a metrological task, we may have a mix of continuous and discrete parameters, like in the agnostic Dolinar receiver of Section 5.3.2. A parameter can be a nuisance; which is an unknown parameter that needs to be estimated, on which we however do not evaluate the precision of the procedure because we are not directly interested in its value. An example of this is the fluctuating optical visibility of an interferometer when we are only interested in the phase. Estimating the nuisance parameters is often necessary/useful to estimate the parameters of interest.

5.2.2 Measurement on the probe

To obtain some information on $\boldsymbol{\theta}$ it is necessary to perform a measurement on the encoded probe $\rho_{x,\boldsymbol{\theta}}$, which will be represented by a POVM $\mathcal{M} := \{M_y^x\}$,

where x are the control parameters and y is the measurement outcome. For the purpose of keeping the notation simple we indicate with x both the controls of the evolution and of the measurement. The probability of obtaining y can be computed from the Born rule and it is

$$p(y|\boldsymbol{\theta}, x) := \mathrm{tr}\left(M_y^x \rho_{\boldsymbol{\theta},x}\right) \ . \tag{5.1}$$

If the measurement is projective, then we end up in a known state and we have extracted the maximum possible amount of information from $\rho_{x,\boldsymbol{\theta}}$. The probe is then reinitialized in the reference state ρ, encoded, and measured again, with the outcome probability given by same expression in Eq. (5.1). If the measurement is weak (meaning non-projective), then there is still information on $\boldsymbol{\theta}$ encoded in the probe state and we do not reinitialize it. The probe may or may not undergo the evolution $\mathcal{E}_{x',\boldsymbol{\theta}}$ again, possibly with different controls x'. After that the probe is measured again using a different POVM $\mathcal{M}' := \{M_{y'}^{x'}\}$, leading to the outcome y'. This procedure can be iterated multiple times, until a projective measurement is performed on the probe, and its state is reinitialized. For a weak measurement the Born rule prescribes an outcome probability that depends on the whole trajectory of previous controls and measurement outcomes. Let us indicate with $\boldsymbol{x}_t := (x_0, x_1, \ldots, x_t)$ and $\boldsymbol{y}_t := (y_0, y_1, \ldots, y_t)$ the tuples containing respectively the controls and outcomes up to the t-th iterations. The probability of obtaining y_{t+1} at the $t+1$-th step is

$$p(y_{t+1}|\boldsymbol{x}_{t+1}, \boldsymbol{y}_t, \boldsymbol{\theta}) := \mathrm{tr}\left(M_{y_{t+1}}^{x_{t+1}} \rho_{\boldsymbol{x}_t,\boldsymbol{y}_t,\boldsymbol{\theta}}\right) \ . \tag{5.2}$$

The case of a continuous measurement can be simulated by taking the appropriate limits, but it is beyond the scope of this work.

5.2.3 Bayesian estimation and particle filter

In the domain of Bayesian estimation we start from a *prior distribution* $\pi(\boldsymbol{\theta})$ for the parameters $\boldsymbol{\theta}$ and update it step by step with the information coming from the measurements, thereby constructing the so called posterior distribution $P(\boldsymbol{\theta})$. We employ the particle filter method [DM97, AMGC02, LC98] (PF) to represent the posterior distribution as an ensemble of points $\{\boldsymbol{\theta}_j\}_{j=1}^N$ in the parameter space Θ, with each point having an assigned weight $\{w_j\}_{j=1}^N$, with N being the number of particles. Fundamentally, we are approximating the posterior distribution with a sum of δ-functions, i.e.

$$P(\boldsymbol{\theta}) \simeq \sum_{j=1}^{N} w_j^t \delta(\boldsymbol{\theta} - \boldsymbol{\theta}_j) \ , \tag{5.3}$$

At the beginning the particles are sampled from $\pi(\boldsymbol{\theta})$ and the weights are initialized to $w_j = \frac{1}{N}$. The update of the posterior to account for new information becomes an update of the weights. From the PF, the estimator $\widehat{\boldsymbol{\theta}} \in \Theta$ for $\boldsymbol{\theta}$ is computed, which in our application is either the mean of the posterior or the most likely value for $\boldsymbol{\theta}$. In case the measurements on the quantum probe are weak (as opposed to projective), it is also necessary to keep track of the measurement backreaction for each possible value of the unknowns $\boldsymbol{\theta}$. In this section we give only an overview of the method, some technical details can be found in Appendix B.1. These details include the resampling scheme, the case where weak (non-projective) measurements are used, and some notes on multimodal posterior distributions.

Bayesian update

If we perform at each step a projective measurement on the probe, then the probability of observing the outcome y, given the control x and the true value $\boldsymbol{\theta}$ of the unknown parameters, is reported in Eq. (5.12). To recover the value of $\boldsymbol{\theta}$ we apply the principles of Bayesian estimation, that is, starting from the prior $\pi(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$, we calculate the posterior probability distribution with the Bayes rule, i.e.

$$P(\boldsymbol{\theta}|x,y) = \frac{p(y|x,\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{P(y)} = \frac{p(y|x,\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\int p(y|x,\boldsymbol{\theta})\pi(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}}\,. \tag{5.4}$$

The denominator is just the normalization required for $P(\boldsymbol{\theta}|x,y)$ to be a probability density. For a series of measurements we apply repeatedly the Bayes rule by using the posterior computed at the previous step as the prior of the next one. Given the tuple of controls $\boldsymbol{x}_{t+1}$ and of outcomes $\boldsymbol{y}_{t+1}$, we can compute the posterior at the $t+1$-th step from posterior at the t-th step with the formula

$$P(\boldsymbol{\theta}|\boldsymbol{x}_{t+1},\boldsymbol{y}_{t+1}) = \frac{p(y_{t+1}|x_{t+1},\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{x}_t,\boldsymbol{y}_t)}{\int p(y_{t+1}|x_{t+1},\boldsymbol{\theta})P(\boldsymbol{\theta}|\boldsymbol{x}_t,\boldsymbol{y}_t)\,\mathrm{d}\boldsymbol{\theta}}\,. \tag{5.5}$$

Notice that for each measurement the probability of obtaining y_{t+1} as a result is independent on the outcomes and controls up to that point and depends only on x_{t+1}. This is precisely because of the reinitialization of the probe after the projective measurements. In order to perform efficiently the Bayesian update on a computer we use the particle filter method (PF), that is, we represent the posterior distribution with a discrete set of points in the space Θ of the parameters, each with its own weight. Fundamentally this means,

we approximate the posterior distribution with a sum of δ-functions, i.e.

$$P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t) \simeq \sum_{j=1}^{N} w_j^t \delta(\boldsymbol{\theta} - \boldsymbol{\theta}_j) \ , \tag{5.6}$$

where the values $\{\boldsymbol{\theta}_j\}_{j=1}^N$ are called particles and $\{w_j^t\}_{j=1}^N$ are the weights at the step t. The values of the particles are to be sampled from the initial prior $\pi(\boldsymbol{\theta})$, while the weights are initialized uniformly on all the particles, i.e. $\omega_j^0 := \frac{1}{N}$. The weights depend on the step because of the Bayesian update of the posterior in Eq. (5.5), which on w_j^t corresponds to the transformation

$$w_j^{t+1} = \frac{p(y_{t+1}|x_{t+1}, \boldsymbol{\theta}_j) w_j^t}{\sum_{j=1}^N p(y_{t+1}|x_{t+1}, \boldsymbol{\theta}_j) w_j^t} \ , \tag{5.7}$$

The particles $\{\boldsymbol{\theta}_j\}_{j=1}^N$ should also depend on the step t, in fact we will introduce a resampling procedure that when triggered generates a new set of particles, which therefore don't necessarily remain unvaried along the estimation. Nevertheless for notational simplicity we avoid putting a time index on $\boldsymbol{\theta}_j$. We indicate to the set of particles and the weights with $\mathfrak{p}_t := \{\boldsymbol{\theta}_j, w_j^t\}_{j=1}^N$, which we call the PF ensemble.

Moments of the posterior

Computing the first moments of the posterior (the mean value and the covariance matrix) corresponds to simple linear algebra operations on the PF ensemble, i.e.

$$\widehat{\boldsymbol{\theta}}_t := \int \boldsymbol{\theta} P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t) \, \mathrm{d}\boldsymbol{\theta} \simeq \sum_{j=1}^{N} w_j^t \boldsymbol{\theta}_j \ , \tag{5.8}$$

and

$$\Sigma_t := \int (\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t)(\boldsymbol{\theta} - \widehat{\boldsymbol{\theta}}_t)^\intercal P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t) \, \mathrm{d}\boldsymbol{\theta} \simeq \sum_{j=1}^{N} w_j^t (\boldsymbol{\theta}_j - \widehat{\boldsymbol{\theta}}_t)(\boldsymbol{\theta}_j - \widehat{\boldsymbol{\theta}}_t)^\intercal \ . \tag{5.9}$$

The mean value of the posterior $\widehat{\boldsymbol{\theta}}$ is our estimator for all continuous parameters throughout the paper. As the estimation proceeds the weights typically concentrate on few particles, while all the others do not play any role in the estimation if not consuming memory. The precision is limited by the average distance between the points $\boldsymbol{\theta}_j$, which depends on the prior $\pi(\boldsymbol{\theta})$ and on the number of particles N. We see in the next section how the introduction of a resampling scheme can mitigate this issue by extracting a new set of particles

$\{\boldsymbol{\theta}'_j\}_{j=1}^N$, which should be in the region where the posterior distribution is concentrated. This means that the density of particles in this region increases allowing for more resolution in distinguishing close values of $\boldsymbol{\theta}$. Throughout the paper we use the same symbols for the "theoretical" moments of the posterior (which are not accessible) and the approximation of these quantities computed from the PF. It will be clear from the contex which quantities we are referencing when.

5.2.4 Controlling agent

A "summary" of the information contained in the Bayesian posterior represented by the PF, such as the mean and covariance matrix of the distribution $P(\boldsymbol{\theta})$, is provided to an agent, like a neural network (NN). This agent then outputs the controls. It is essential for the agent to be specifically trained for the experiments it is intended to optimize. This means, for instance, that precise values of the decoherence rates and visibilities should be known and incorporated into the simulation, unless they are included among the parameters $\boldsymbol{\theta}$ to be estimated. In this manner, the knowledge on $\boldsymbol{\theta}$, gained through measurements, can be adaptively leveraged to control both the evolution and the measurements performed on the probe through the agent, with the aim of maximizing the final precision of the estimation. We envision carrying out experiments with a small trained agent programmed on fast hardware, like a Field Programmable Gate Array (FPGA), located in the proximity of the experiment.

5.2.5 The precision-resources paradigm

In our framework each measurement performed on the probe consumes some amount r of a specific "resource", which is costly in the context of the experiment and must be defined by the user, according to the limitation of the setup. Once the total available resources R are depleted, the estimation is concluded, and the final value of the estimator $\widehat{\boldsymbol{\theta}}$ is computed. Some examples of resources are the total estimation time, used for the NV center platform, the average number of consumed photons, or the amplitude of a signal, like in the Dolinar receiver. For the optimization of the metrological task the definition of the resource is as important as the precision figure of merit. There is no right or wrong resource in an estimation task, it depends on the experimentalist's choices and on their understanding of the laboratory limitations in the implementation of the task.

5.2.6 The measurement loop

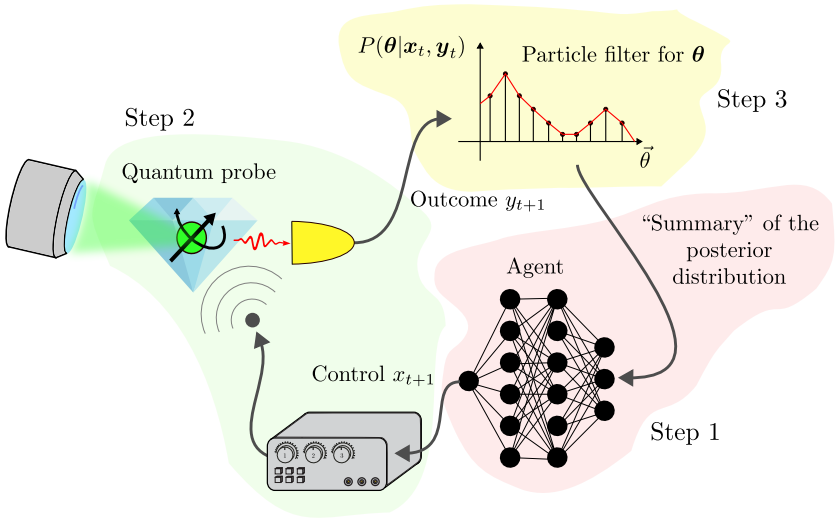


Figure 5.1: Schematic representation of the three steps of information flow within the measurement loop. The labels refer to the $(t+1)$-th iteration. In the first step (pink region of the figure), the summary information computed from the particle filter is fed into the agent (here represented as a NN) which determines the control parameters for both the evolution and the measurement of the probe in this iteration, collectively represented by the variable x_{t+1}. In the second step (green region) the parameters $\boldsymbol{\theta}$ are encoded in the probe state and the measurement is conducted, producing the outcome y_{t+1}. In the third step (yellow region) this outcome is input into the particle filter leading to the update of the Bayesian posterior distribution on the parameters $\boldsymbol{\theta}$ and on state of the probe (if applicable).

The metrological task is simulated as a loop of consecutive operations, which we call the *measurement loop*, represented in Fig. 5.1. Within this loop, for each iteration numbered from $t = 0$ to $M - 1$, a single measurement is performed. We proceed by describing the generic iteration of the loop (let it be the $t+1$-th iteration), which is comprised of three steps. As described in the caption of Fig. 5.1 we indicate with the symbol x_{t+1} the controls produced by the agent for the evolution of the probe and its measurement, while under y_{t+1} we understand the outcome of the measurement, both taken at the $t+1$-th iterations of the loop. The objects $\boldsymbol{x}_t := (x_0, x_1, \ldots, x_t)$ and $\boldsymbol{y}_t := (y_0, y_1, \ldots, y_t)$ are tuples that contains the controls and the measurement outcomes up to the time t. The distribution $P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t)$ is the Bayesian posterior updated with the outcomes up to step t of the measurement loop.

1. In the case of adaptive strategies, the choice of x_{t+1} operated by the agent shall be represented without loss of generality via the mapping

$$x_{t+1} = \mathcal{F}_{\boldsymbol{\lambda}}\{P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t); \boldsymbol{y}_t; R_t; t\} \ , \tag{5.10}$$

where, defining r_j the resource consumption at the j-th step of the protocol, we compute the total resource consumed up to the t-th step as $R_t := \sum_{j=0}^{t} r_j$. Non-adaptive strategies are described by maps $\mathcal{F}$ that carry no functional dependence upon $P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t)$ or $\boldsymbol{y}_t$, i.e

$$x_{t+1} = \mathcal{F}_{\boldsymbol{\lambda}}\{R_t; t\} \ . \tag{5.11}$$

The mapping $\mathcal{F}_{\boldsymbol{\lambda}}$ depends on the trainable variables of the strategy, collectively indicated with $\boldsymbol{\lambda}$, that are later optimized. These are the weights and biases for a NN. For the non-adaptive strategies of this work the agent is just a list of controls which are applied sequentially in the measurement loop, and $\boldsymbol{\lambda} = \boldsymbol{x}_{M-1}$. For all the examples the NN has by default 5 hidden layers with 64 neurons each, and the activation function is tanh, which has been proved to be good for approximating smooth functions [DRLM21].

2. Suppose that the measurements are projective, and that the probe's state is reinitialized after each iteration. Then the probability of observing the outcome y_{t+1} at the $t+1$-th step is given by $p(y_{t+1}|x_{t+1}, \boldsymbol{\theta})$, which is computed from the Born rule according to the known quantum dynamics of the probe that has been coded in the simulation. This probability, which we henceforth call the "model", depends only on the controls x_{t+1} and on the encoded parameters to estimate $\boldsymbol{\theta}$. At this second step of the measurement loop, the outcome y_{t+1}, which is a stochastic variable, is extracted from the model distribution, i.e.

$$y_{t+1} \sim p(y_{t+1}|x_{t+1}, \boldsymbol{\theta}) \ . \tag{5.12}$$

If the probe is subject to a weak measurement, then the outcome probability depends on the whole sequence of previous controls and outcomes, because of the measurement backreaction, i.e.

$$y_{t+1} \sim p(y_{t+1}|\boldsymbol{x}_{t+1}, \boldsymbol{y}_t, \boldsymbol{\theta}) \ . \tag{5.13}$$

3. The observation of y_{t+1} is then incorporated into the posterior through the Bayes rule, i.e.

$$P(\boldsymbol{\theta}|\boldsymbol{y}_{t+1}, \boldsymbol{x}_{t+1}) \propto p(y_{t+1}|x_{t+1}, \boldsymbol{\theta}) P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t) \ . \tag{5.14}$$

At the first iteration the prior $\pi(\boldsymbol{\theta})$ appears instead of the posterior. If the measurements are weak, then the model probability has the form reported in Eq. (5.13).

The stopping condition of the measurement loop can be trivial, i.e. we assign a maximum number of iterations M, or based on the amount of resources available, i.e. it can be a limit on R_t.

5.2.7 Training with model-aware reinforcement learning

The figure of merit for the precision depends on the type of metrological task. In the examples concerning the NV center platform, where the parameters $\boldsymbol{\theta}$ are continuous, the mean square error (MSE) is used, i.e. the loss for a single estimation is

$$\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta}) := \mathrm{tr}\left[G \cdot (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})^{\intercal}\right] , \tag{5.15}$$

with $G \geq 0$ being a positive semidefinite weight matrix, and $\widehat{\boldsymbol{\theta}}$ being the mean of the posterior. The weight matrix G controls which errors contribute to the loss $\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})$ and how much. It discriminates therefore between parameters of interest and nuisances, with the latter having the corresponding entries in the G matrix set to zero. In a discrete estimation task, illustrated later in this work for a photonic platform), both $\boldsymbol{\theta}$ and $\widehat{\boldsymbol{\theta}}$ are discrete, i.e. $\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}} \in \Theta = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_k\}$. Accordingly, the loss for a single instance of the task is expressed in terms of a Kronecker delta, i.e.

$$\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta}) := 1 - \delta(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta}) , \tag{5.16}$$

with

$$\widehat{\boldsymbol{\theta}} := \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t) , \tag{5.17}$$

being the maximum *a posterior* estimator. Optimizing the control strategy entails identifying the agent that minimizes the average loss $\mathbb{E}[\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})]$, averaged over all possible choices of $\boldsymbol{\theta}$ and over all the stochastic processes involved in the estimation of $\boldsymbol{\theta}$, see Appendix B.3. Each potential agent is characterized by the values of a set of trainable variables, denoted as $\boldsymbol{\lambda}$, that influence the individual losses of the problem as well as the associated $\mathbb{E}[\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})]$. The optimal strategy can hence be abstractly identified with the value $\boldsymbol{\lambda}^{\star} := \mathrm{argmin}_{\boldsymbol{\lambda}} \mathbb{E}[\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})]$ that minimizes the average loss. The training of the agent is an iterative algorithm that aims to discover a strategy

closely approximating the performance of such optimal $\boldsymbol{\lambda}^\star$ via a sequence TS(1), TS(2), $\cdots$, TS(I) of recursive updates,

$$\boldsymbol{\lambda}_0 \overset{\text{TS}(1)}{\longrightarrow} \boldsymbol{\lambda}_1 \overset{\text{TS}(2)}{\longrightarrow} \cdots \overset{\text{TS}(I)}{\longrightarrow} \boldsymbol{\lambda}_I \simeq \boldsymbol{\lambda}^\star \,, \tag{5.18}$$

with $\boldsymbol{\lambda}_0$ being an initial educated guess. The construction of the learning trajectory Eq. (5.18) relies on the possibility of computing an estimation $\mathcal{L}(\boldsymbol{\lambda})$ of the average loss $\mathbb{E}[\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})]$ associated with a generic agent $\boldsymbol{\lambda}$. This is typically done simulating in parallel B estimations of randomly selected values $\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_B$ of the parameters $\boldsymbol{\theta}$. Accordingly we can then write

$$\mathcal{L}(\boldsymbol{\lambda}) := \frac{1}{B}\sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k) \simeq \mathbb{E}[\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})] \,, \tag{5.19}$$

where $\ell(\widehat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k)$ represents the local loss of the k-th estimation which inherits a functional dependence upon $\boldsymbol{\lambda}$ from the multiple operations that the agent has to perform in order to recover $\boldsymbol{\theta}_k$. Exploiting such dependence we can compute the gradient $\mathcal{G}(\boldsymbol{\lambda}) := \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\mathcal{L}(\boldsymbol{\lambda})$ of $\mathcal{L}(\boldsymbol{\lambda})$ via automatic differentiation (AD), running in reverse through all the operations of the measurement loop. The upgrade of the agent parameters is performed at each training step with stochastic gradient descent through the formula

$$\boldsymbol{\lambda}_i \overset{\text{TS}(i+1)}{\longrightarrow} \boldsymbol{\lambda}_{i+1} = \boldsymbol{\lambda}_i - \alpha\mathcal{G}(\boldsymbol{\lambda}_i) \,, \tag{5.20}$$

with $\alpha \in (10^{-4}, 10^{-1})$ being the learning rate. Actually, in the examples reported, the Adam [KB15] optimizer is used, which prescribes a more complicated update step, conceptually similar to Eq. (5.20). Also we use a learning rate decreasing with the training step i. Since the derivatives are propagated through the model for the sensor in Eq. (5.12), this training is a form of model-aware policy gradient reinforcement learning. The gradient descent training of $\boldsymbol{\lambda}$ will converge to a minimum of the loss, however, we don't have any guarantee that it will be $\boldsymbol{\lambda}^\star$. Since the loss is defined in terms of the stochastic outcomes $\boldsymbol{y}_t$, special precautions are necessary to compute an unbiased estimator for its gradient [PPM23], which involve the addition of the log-likelihood terms $\log p(y_{t+1}|\boldsymbol{x}_{t+1}, \boldsymbol{y}_t, \boldsymbol{\theta})$ to the loss. See the Appendix B.3 for details on the loss definition and on its gradient. When doing an estimation with a fixed number of measurements $M_{\max}$ or a fixed maximal amount of resources $R_{\max}$, choosing a loss $\mathcal{L}(\boldsymbol{\theta})$ that is sensitive only to the performances of the estimator $\boldsymbol{\theta}$ at the very end of the simulations doesn't necessarily produce the optimal strategies for $M < M_{\max}$, $R < R_{\max}$. The simplest solution would be to repeat the optimization for each smaller $R_{\max}$ we want

to characterize. There is, however, a way to find an approximate solution $\forall R \leq R_{\max}$, which requires just a training that optimizes the cumulative loss instead of Eq. (5.19), i.e.

$$\mathcal{L}_{\text{cum}}(\boldsymbol{\lambda}) := \frac{1}{M_{\max} B} \sum_{t=0}^{M_{\max}-1} \sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) \ . \tag{5.21}$$

This loss has the effect to pressure the agent to learn a strategy that is optimal $\forall\, R \leq R_{\max}$, and it has been used in the examples on the NV center platform. A further version of the loss is the logarithmic loss, which prescribes the use of the logarithm of the average loss on the batch instead of the average loss in Eq. (5.21). See Section B.3.5 for more details.

5.2.8 Differentiability of the particle filter

The main ingredient of our approach is the combination of particle filter Bayes updates with model-based reinforcement learning. This represents a challenge, since PF updates involve steps where it is not immediately obvious how they could be made differentiable for gradient computation. As the estimation proceeds, the weights of the PF get concentrated on few particles only. To optimize the memory usage we implement a resampling procedure, that, when called, extracts a new sets of particles $\{\boldsymbol{\theta}'_j\}_{j=1}^N$ according to the posterior $P(\boldsymbol{\theta})$ and resets the weights to $w'_j = \frac{1}{N}$. This resampling procedure consists of three steps that can be toggled on and off at will. These are: the resampling from the posterior $P(\boldsymbol{\theta})$, the perturbation of the newly extracted particles, and the proposal of new particles. We have optimally combined them through a trial-and-error procedure, see Section B.1.1. All these steps involve the extraction of discrete stochastic variables, an operation that in principle is not differentiable and would completely impede the propagation of the gradient later needed for reinforcement learning. While the last two steps can be trivially made differentiable with the reparametrization trick (see Section B.2.1), for circumventing the issue of resampling the discrete PF ensemble, we could modify the loss by adding the log-likelihood of the stochastic outcomes as we do for the measurements. However, for a large number of particles N, this would affect negatively the variance of the estimated gradient in the simulations. Instead, we use importance sampling to extract the new particles from a distribution $Q(\boldsymbol{\theta})$ different from the posterior and we set the new weights proportional to the factor $\frac{P(\boldsymbol{\theta})}{Q(\boldsymbol{\theta})}$, so that the PF always represents the posterior [KHL18]. In this way the gradient can propagate through a resampling event via the term $P(\boldsymbol{\theta})$ in the weights. Along with the importance sampling we have implemented correction introduced by

Ścibior and Wood [SW21] to get differentiable resampling, and we proved its efficacy for the mean square error loss, see Section B.2.2. This correction is complementary to importance sampling and its effect is to add to the loss the least possible numbers of log-likelihood terms for particle extraction events, so not to compromise the stability of the training. The Bayes rule, being just the product of the model probability and the previous posterior, is trivially differentiable.

5.3 Results

In this section we present two application of this technique, to static field magnetometry with NV center and to quantum communication with the Dolinar receiver.

5.3.1 DC magnetometry with NV centers

The nitrogen-vacancy (NV) centre in diamond is a point defect that enables initialisation, detection and control of its electronic spin, featuring very long quantum coherence time, even at room temperature. As such, it has been used in applications such as magnetometry, thermometry, and stress sensing [Gal19, DDF22, CMZ+18, Maz10, BSB+20]. The electronic spin is sensitive to magnetic fields; for example static fields determine the electron Larmour frequency, which can be measured as an accumulated phase by a Ramsey experiment. This experiments are realizes by applying two $\pi/2$ pulse to the spin, followed by illumination with green light and detection of the photoluminescence. A single measurement has a binary outcome, yielding ± 1 with probabilities

$$p(\pm 1|\omega, T_2^\star, \tau) := \frac{1}{2} \pm \frac{1}{2} e^{-\tau/T_2^\star} \cos(\omega\tau) \ . \tag{5.22}$$

The free evolution time τ is controlled by a trainable agent, while $\omega := \gamma B$ represents the unknown precession frequency to be estimated, which is proportional to the static magnetic field B with $\gamma \simeq 28\,\mathrm{MHz/mT}$. The parameter $T_2^\star$ denotes the transverse relaxation time, serving as the time scale for the dephasing induced by magnetic noise. The optimization of the NV center as a magnetometer has been extensively studied in the literature with analytical tools [SGL+21, FGC13], with numerics [DAM20, MDB21, GFWC12, ORC+22, CGAB21, BBD+16, SGK+19, ZHR+23, NMD12, WPS+17, DBG+19, BB17, FGC12], and with Machine Learning [LCL+20, FSB21, TIO+22]. We conducted multiple estimations over the same parameter ranges chosen in

the work of Fiderer *et al.* [FSB21], in order to allow an easy comparison of the results. The prior for the frequency ω is uniform in $(0, 1)$ MHz. Fig. 5.2 compares the performances of the optimized adaptive (NN) and non-adaptive strategies against the Particle Guess Heuristic (PGH) [WGFC14], which is a commonly referenced strategy in the literature. Additionally, we introduced a variant of the σ^{-1} strategy [FGC13], named $\sigma^{-1}\&T^{-1}$, which accounts for the finite coherence time. According to the $\sigma^{-1}\&T^{-1}$ strategy, the next evolution time τ is computed from the covariance matrix Σ of the current posterior distribution as $\tau = \left[\mathrm{tr}(\Sigma)^{\frac{1}{2}} + 1/T_2^\star\right]^{-1}$. For computing the controls of the PGH strategy, two particles $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are drawn from the particle filter; the evolution time is then computed as $\tau = (||\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2||_2 + \varepsilon)^{-1}$ with $\varepsilon := 10^{-5}\,\mu\mathrm{s}^{-1}$.

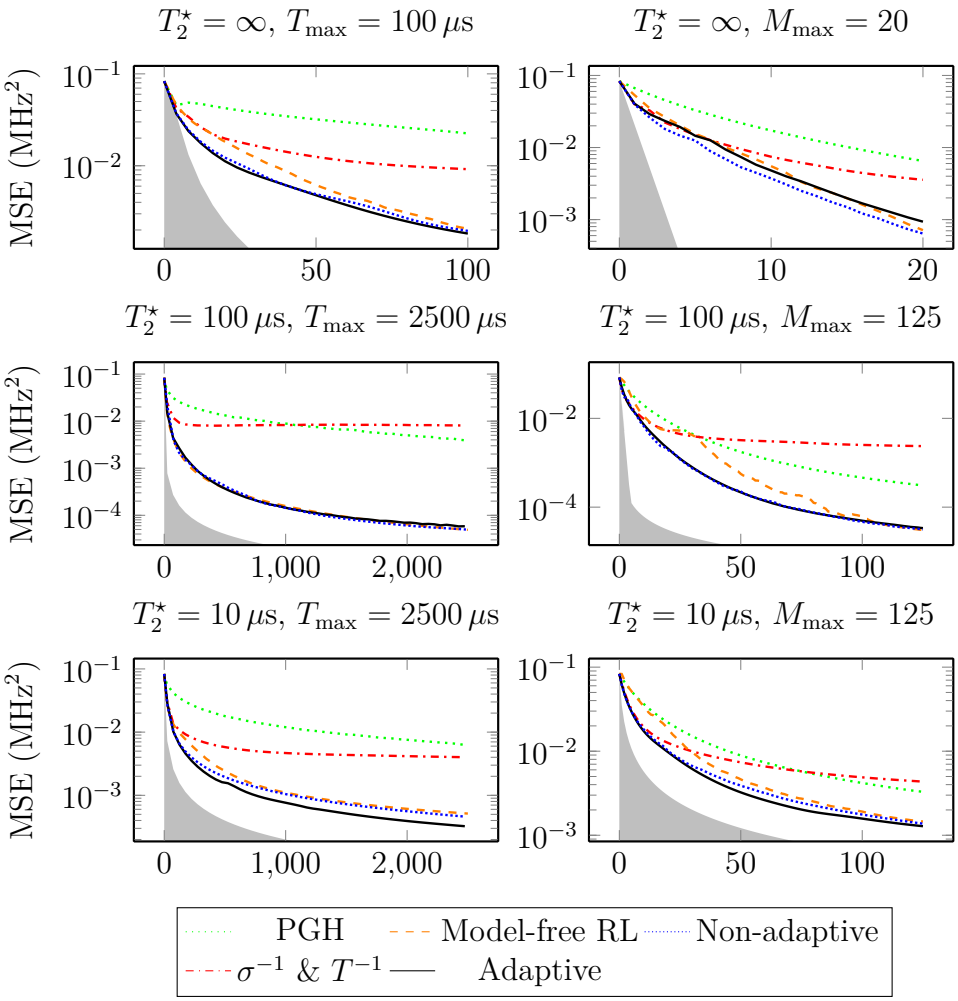


Figure 5.2: These plots refer to static field magnetometry with an NV center, executed in different conditions. The MSE on ω is plotted as a function of the total number of consumed resources, which are either the maximum number of measurements $M_{\max}$ or the maximum total free evolution time of the probe, i.e. $T_{\max} \geq \sum_{t=0}^{M-1} \tau_k$. The adaptive (NN) and non-adaptive strategies are optimized using model-aware RL. The description of the σ^{-1}&T^{-1} and PGH strategies can be found in the main text. With the label "Model-free RL" we denote the performances obtained in [FSB21] with model-free RL, which are never better than the non-adaptive strategy optimized with our techniques. The shaded grey area represent the (non-tight) ultimate precision bound, computed either from the Cramér-Rao bound or from a bit-counting arguments, see the Supplemental Material of [BZMG]. The title of each plot contains the transverse relaxation time $T_2^\star$, which is the time scale of the dephasing noise, along with the maximum amount of resources used in the simulations. The number of particle in the PF was $N = 480$ for the first and third rows, and $N = 1024$ (left) and $N = 1536$ (right) for the second row, which are significantly smaller numbers than the ones used in [FSB21]. The weights and biases of the NN have been initialized randomly, while the initial τ for the non-adaptive approach is a deterministic linear ramp increasing after each measurement.

For each plot, the best between our optimized adaptive and non-adaptive strategies, outperforms all the other approaches, as illustrated in Fig. 5.2. There are two comparisons to be made: on one hand we have the optimized adaptive vs. the non-adaptive strategies, which are both original results of this work; on the other hand we have model-free vs model-aware RL, where application of the latter to NV center magnetometry has been studied in [FSB21]. We shall start with the first comparison. Notably, the optimal results for extended coherence times ($T_2^\star = 100\,\mu\text{s}, \infty$) are achieved using non-adaptive strategies, which offer several practical advantages in the experimental implementation. Primarily, since the controls are fixed *offline* before the experiment, there's no requirement for real-time feedback via rapid electronics. Furthermore, there's also no need to update the Bayesian posterior on the fly given the absence of adaptivity. Instead, the measurement outcomes can be processed offline, post-measurement, using more powerful hardware. This would significantly reduce online memory usage as there's no need for real-time updates to the particle filter. In the third row of Fig. 5.2 we see a gap between the performances of "Adaptive" and "Non-adaptive", and in [BZG24b] we give more examples of the adaptivity being useful. Regarding the second comparison of model-aware and model-free RL, we observe that no strategy trained with model-free RL can beat even the non-adaptive strategy, which means that the results of [FSB21], although close to ours, cannot prove that the NN has been trained to exploit adaptivity, and that it hasn't simply learned an optimal non-adaptive sequence of measurement times τ. Beside this, we notice that our "Adaptive" strategy and the "Model-free" approach give results that are closer toward the end of the estimation, while they differ for intermediate times. This is due to our use of the cumulative loss. In the simulation with $T_2^\star = \infty$, $M_{\text{max}} = 20$, the NN strategy performs worse than the non-adaptive one because it remains stuck in a local minimum during the training. In Fig. 5.3 we reported five examples of optimal adaptive trajectories for the estimation of $\omega = 0.2\,\text{MHz}$ referring to $T_2^\star = 10$, together with the optimal non-adaptive strategy. We observe that multiple runs of the agent training will produce consistent performance but not necessarily the same optimized agent. In conclusion we want to put forward an explanation to why the adaptive control seems to give so little advantage with respect to the optimized non-adaptive strategy. For the adaptivity to be advantageous, the phase $\omega\tau$ must be known to some extent. As the error on ω goes down, the evolution time increases, so that the uncertainty on $\omega\tau$ doesn't go to zero even after many measurements, which leaves very little room to adaptivity for improving the estimation precision.

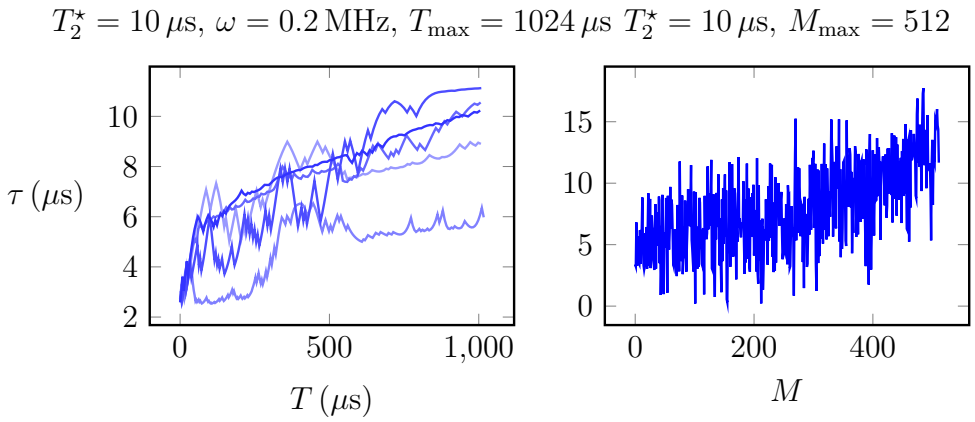


Figure 5.3: Control strategies for the estimation of ω in DC magnetometry. For the time limited estimation (on the left) five example trajectories produced by the NN are plotted for $\omega = 0.2\,$MHz. On the right, the optimal non-adaptive strategy for the measurement-limited estimation is presented. This prescribes a growing τ with a random pattern superimposed.

5.3.2 Agnostic Dolinar receiver

Consider the challenge of distinguishing between two known coherent states, $|-\alpha\rangle$ and $|\alpha\rangle$, where $\alpha \in \mathbb{R}$ and $\alpha > 0$, using a single copy of the signal $|\pm\alpha\rangle$. The Dolinar receiver optimally addresses this problem through linear optics and photon counting [Dol73b, Ger04a, ITF+12, ADPP11b, CMG07b, PL14, TSvLL05b]. For this device multiple Machine Learning approaches can be found in the literature [BRYC20b, CHH+22]. In some recent studies [SGA15b] (and also Chapter 4), a variant of this device was introduced which doesn't require a local oscillator (LO) on the receiver side, which must be in phase with the sender's laser. This is the agnostic Dolinar receiver, in which in place of the LO, n copies of $|\alpha\rangle$, called the reference states, are sent to the receiver from the sender, alongside the signal $|\pm\alpha\rangle$. We furthermore assume that classical knowledge about the state $|\alpha\rangle$ is missing, i.e. α is an unknown parameter of the estimation. In Fig. 5.4 we represent schematically this device, which leverages the states $|\alpha\rangle^{\otimes n}$ to perform the discrimination task on the sign of the signal $|\pm\alpha\rangle$. The signal $|\pm\alpha\rangle$ enters from the left and is sequentially combined with one of the reference states $|\alpha\rangle$ on a programmable beam splitter with adjustable reflectivity θ_i. At each beam splitter, one of the two ports undergoes measurement by a photon-counter, while the residual signal $|\psi_i\rangle$ from the other port is fed forward to the subsequent beam splitter. The photon counting result is used to update the Bayesian posterior on α and

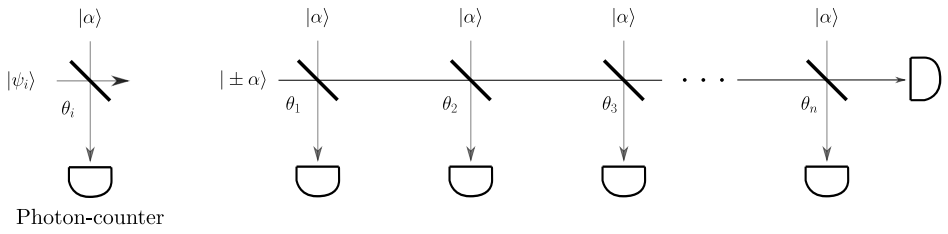


Figure 5.4: Schematic representation of the agnostic Dolinar receiver: each thick diagonal line symbolizes a beam splitter with programmable reflectivity θ_i. Each "D" device denotes a photon-counter. On the left side of the figure, the building block of this apparatus is illustrated. Here, the input state at step i, named $|\psi_i\rangle$, is combined with one of the n training states $|\alpha\rangle$. One of the two ports undergoes measurement via photon-counting. At the device's end the second output port is also measured, ensuring no information is left unused.

on the signal's sign, from which the reflectivity for the upcoming beam splitter is determined via a NN. In this task, which is a combination of estimation and discrimination, there are two undetermined parameters: one continuous, i.e. the signal's amplitude $\alpha \in \mathbb{R}$, and one discrete, i.e. the signal's sign. The receiver's performance is assessed based on the error probability in the task of signal classification, with the loss being the one in Eq. (5.16), while the amplitude α is a nuisance parameter. See [BZG24b] for the details about the loss and the input to the NN.

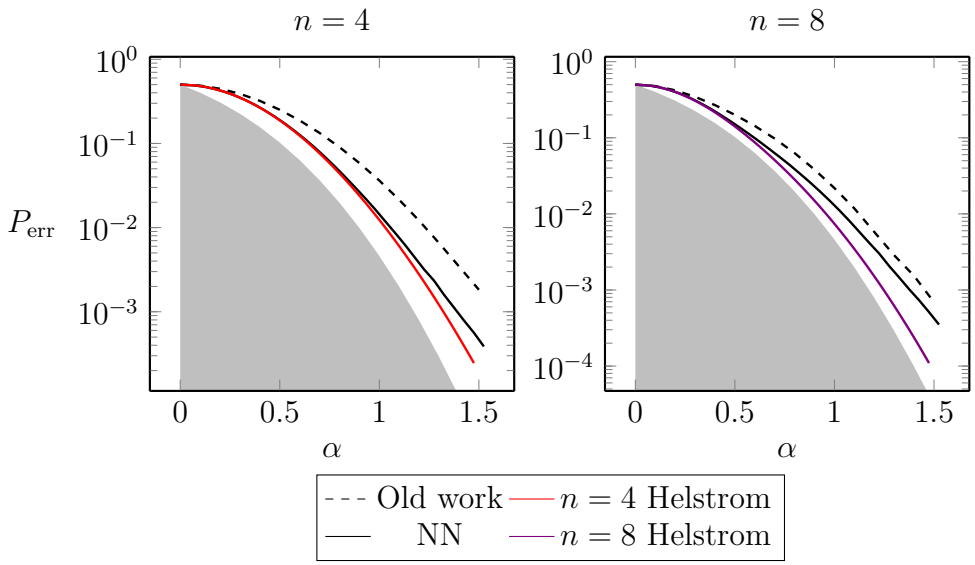


Figure 5.5: Comparison of error probabilities for various strategies with different numbers of copies of $|\alpha\rangle$, specifically $n = 4$ and $n = 8$. The shaded gray area is the region excluded by the Helstrom bound [Hel69, Hol73b], which is the lowest error probability theoretically achievable when assuming having an infinite number of reference states ($n = \infty$) at disposal. The solid red and violet lines are the Helstrom bound calculated for a finite number of copies of $|\alpha\rangle$, respectively $n = 4$ and $n = 8$. These lines are the exact replicas of the line named "Heterodyne" that can be seen in Fig. 4.6. For the details on the computation of the Helstrom bound see [BZG24b]. The black dashed line showcases the lowest error found in the old work [ZPFG21], without Machine Learning, while the black solid line is the performance achieved using the NN. The performances of the optimal non-adaptive strategies haven't been reported since they can't rival the ones of the NN. For both the training and the performances evaluation we used $N = 512$ particles. The weights and biases of the NN have been initialized randomly.

The simulation results are presented in Fig. 5.5. We compared the performances of our adaptive procedure with the current state-of-the-art solution for this problem [ZPFG21]. In each scenario, we achieved superior results with the NN. Notably, we nearly reached the theoretical bound in our primary area of interest, relevant for long-distance communications, i.e. the full quantum limit with $\alpha \lesssim 1$, and a small number of reference states (specifically, $n = 4$). For large α, the error probability is already very small and we are in the classical limit.

5.4 Conclusions

Overall, our research highlights the benefits of merging Machine Learning with modern quantum technologies. We introduced a framework, complemented by a versatile library, capable of addressing a wide spectrum of quantum parameter estimation and metrology challenges both in the Bayesian and in the frequentist framework, applicable to a plethora of platforms. Our methods have the potential to accelerate the development of practical applications in quantum metrology. The capability to precisely estimate physical parameters through quantum systems could revolutionize numerous sectors, including biology, fundamental physics, and quantum communication. Through the tool of model-aware reinforcement learning, we aspire to catalyse progress in these domains, smoothing the shift of quantum-based metrology from proof-of-principle experiments to industrial applications. The technique of model-aware RL for agent optimization could be in principle applied to a wide range of problems in quantum information, including quantum error correction and entanglement distillation, which would require engineering other losses. The main obstacle of extending this approach in other fields of quantum information beyond metrology is, however, the exploding dimensionality of the quantum systems states that would need to be simulated.

Chapter 6

Conclusions

This PhD thesis drew its attention to addressing several unresolved challenges in quantum information through a combination of theoretical analysis and the application of numerical methods. The primary aim has been to advance the understanding and practical implementation of quantum information systems, specifically by exploring novel approaches to quantum state discrimination and quantum metrology techniques.

The first significant contribution of this thesis involves the introduction of a novel distance metric between quantum states within a variational quantum algorithm. Our findings demonstrate that this new distance metric significantly enhances performance compared to traditional metrics such as trace distance or fidelity, offering a more effective approach to quantum state manipulation in this context.

The first results reported in this thesis are about the application of a novel distance between quantum states within a variational quantum algorithm. Our findings demonstrate that this new distance metric significantly enhances performance compared to traditional metrics such as trace distance or fidelity, thereby offering a more effective approach to quantum error correction.

The second major contribution is a generalization of the Dolinar receiver, a device used for optimal discrimination of coherent states of light. Initially, an analytical approach was undertaken, as detailed in Chapter 4, giving valuable insights but failing to achieve the optimal theoretical performance. Subsequently, a more generalized framework was developed, in Chapter 5, stripping the problem to its fundamental components. This framework allows for extracting information from a quantum system with unknown parameters using weak adaptive measurements. Two specific applications of this framework are presented in this thesis: an analysis of the NV center problem and a more in-depth study of the Dolinar receiver. Both studies significantly advance the current state of the art in their respective fields, with potential direct

applications in quantum communication over noisy channels and quantum metrology. Notably, the new agnostic Dolinar apparatus developed in this research nearly closes the gap to the optimal theoretical performance.

In summary, this thesis has made substantial gains in the field of quantum information by introducing some new techniques and a new framework to solve problems in quantum metrology and sensing. These advancements contribute to the theoretical understanding of quantum systems but also hold promise for practical applications in quantum communication and metrology. The work presented here lays a strong foundation for future research and experimental exploration in these areas.

6.1 Acknowledgments

The work done in this thesis has received support from MIUR via PRIN 2017 (Progetto di Ricerca di Interesse Nazionale): project QUSHIP (2017SRN-BRK) and via project PRO3 Quantum Pathfinder.

Ringraziamenti

Il primo ringraziamento va sicuramente al mio relatore, Vittorio Giovannetti, che mi ha seguito fin dalla laurea triennale ed è stato decisamente più paziente e comprensivo del dovuto in varie occasioni. Il secondo va sicuramente a Federico Belliardo, un fratello maggiore e maestro di vita, con cui ho lavorato per produrre l'ultimo capitolo di questa tesi, senza di lui non sarebbe stato possibile. Gli altri ringraziamenti vanno a due gruppi di persone, con cui ho costruito i "side projects" che ho coltivato durante i miei anni del PhD. Il gruppo di persone con cui ho lavorato per realizzare la Gara a Squadre di Fisica e lo Stage di Fisica a Pisa merita la mia eterna riconoscenza, avendo lavorato giorno e notte sostanzialmente per fare beneficienza, anche lottando contro chi vuole affondare questi progetti. L'altro gruppo è fibonhack, che mi ricorda sempre quanto ci sia forte correlazione tra "il ruzzare" e il saperne a pacchi del proprio settore.

Infine, grazie alla Scuola Normale, un luogo che coltiva da secoli una comunità incredibile, con alcuni bassi ma con vette stratosferiche. Troppi allievi sono ingrati verso questa istituzione, ma puntualmente non se ne vanno, chissà perché. Questi (quasi) 10 anni sono volati, proprio grazie a questa comunità.

Una volta Normalista, per sempre Normalista.

Appendices

Appendix A

Agnostic Dolinar receiver

A.1 An equivalent description of the Dolinar receiver

Here we review the alternative formulation of the Dolinar receiver presented in [TSvLL05a], which is depicted in Fig. A.1 and comes from the equivalence between a continuous photon counting process and a sequence of beam splitters and photon–detectors [Ban94]. The input state comes in the apparatus from the left and goes through a sequence of very similar steps. Each of the diagonal rectangles is a beam splitter of very small reflectivity $\theta \ll 1$. The input state is mixed with the vacuum $|0\rangle$ via this beam splitter, displaced with the displacement gate $\hat{D}(\gamma_k \sin\theta)$, and then undergoes photon counting. The measurement result is used, in addition to the known value of α, to decide the next displacement parameter γ_{k+1}. Then, the discrimination result will simply be the parity of the total number of photons counted. It can be shown that, with the correct choice of γ_k, and for the number of steps going to infinity, this apparatus tends to the Helstrom bound.

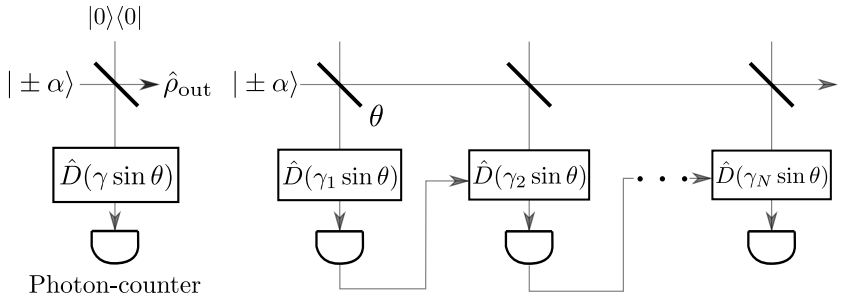


Figure A.1: Discretized description of the Dolinar receiver. The rectangles represents a displacement gate, which is followed by photon counters, while the diagonal lines represent beam splitters. This equivalent schematization is used in Section 5.3.2.

A.2 Derivation of Eq. (4.28)

Observe that the following identities hold

$$\sum_{m=0}^{\infty} \hat{\Pi}_m \left|\sqrt{n}\alpha\right\rangle\!\left\langle\sqrt{n}\alpha\right| \otimes |\pm\alpha\rangle\langle\pm\alpha| \, \hat{\Pi}_m$$

$$= \sum_{m=0}^{\infty} \mathfrak{p}(m;\mu) \, |m,\pm\rangle\langle\pm,m| \,, \tag{A.1}$$

$$\sum_{m=0}^{\infty} \| \, \Pi_m(\hat{\rho}_+ - \hat{\rho}_-)\Pi_m \, \|_1 = \tag{A.2}$$

$$\sum_{m=0}^{\infty} \mathfrak{p}(m;\mu)\sqrt{1 - |\, \langle m,+|m,-\rangle \,|^2},$$

which imply Eq. (4.28) by noticing that

$$\langle m,+|m,-\rangle = \left(\frac{n-1}{n+1}\right)^m. \tag{A.3}$$

Appendix B

Qsensoropt

B.1 Implementation of the particle filter

B.1.1 Resampling scheme

While for a small number of unknown parameters we could still obtain good performances even if no resampling procedure is performed, it is essential for larger dimensions. Indeed the density of particles, i.e. the resolution in $\boldsymbol{\theta}$, after the initialization, is inversely proportional to the volume of the parameter space, which grows exponentially in the number d of dimension of the Θ space. To solve this problem it is typical to perform during the estimation a resampling of the particles according to the posterior distribution, which is triggered by the condition

$$N_{\text{eff}} := \frac{1}{\sum_{j=1}^{N}(w_j^i)^2} < r_t N \ , \tag{B.1}$$

where r_t is the resampling threshold that is kept fixed at $r_t = 0.5$ in all the simulations of the paper. The left hand side of Eq. (B.1) is sometimes called the effective number of particles N_{eff}.

Soft resampling

The simplest resampling scheme prescribes the extraction of N samples with repetitions from the set of indexes $J = \{1, \cdots, N\}$, each weighted with the corresponding w_j, $j \in J$. We will call $\phi(j) : J \to J$ the map that gives the outcome of the j-th extraction event. The indexes $j \in J$ corresponding to the particles $\boldsymbol{\theta}_j$ that have large weights are extracted more frequently, while the particles with small weights tend to disappear. In our implementation we considered a slightly more general version of this procedure which goes

under the name of soft resampling [KHL18], that is, we mix the probability distribution represented by the weights $\{w_j\}_{j=1}^N$ with a uniform distribution on $\{\boldsymbol{\theta}_j\}_{j=1}^N$ by constructing the soft-weights q_j defined as

$$q_j := \alpha w_j + (1-\alpha)\frac{1}{N} , \tag{B.2}$$

where $\alpha \in [0,1]$ is a parameter characterizing the effectiveness of the resampling. With $\alpha = 1$ we have the traditional procedure, while with $\alpha = 0$ no actual resampling is performed, because we extract the new particles from a uniform distribution, just like at the beginning. With $\alpha = 0$ the particles with low weights are not cut away from the ensemble but persist after the process. With an intermediate value of α (by default we set $\alpha = 0.5$) only a fraction α of the particles are effective for the resampling, because the other fraction $(1-\alpha)$ is expected to be distributed uniformly. We call $\boldsymbol{\theta}'_j$ the new particles extracted from q_j, i.e.

$$\boldsymbol{\theta}'_j = \boldsymbol{\theta}_{\phi(j)} . \tag{B.3}$$

Their corresponding weights are chosen, so that the ensemble of the PF represents the same distribution as before the resampling. These are

$$w'_j \propto \frac{w_{\phi(j)}}{q_{\phi(j)}} = \frac{w_{\phi(j)}}{\alpha w_{\phi(j)} + (1-\alpha)\frac{1}{N}} , \tag{B.4}$$

that still need to be normalized. With this choice for w'_j the PF represents the correct posterior even though the particles have been sampled from a different distribution. The probability density function represented by the PF is, roughly speaking, proportional to the product of the weights w'_j and the density of particles at the position $\boldsymbol{\theta}'_j$, i.e. $q_{\phi(j)}$, which with our choice for w'_j is exactly $w_{\phi(j)}$, i.e. the weight of the particle $\boldsymbol{\theta}'_j$ prior to the resampling step. In the next section we detail this relation. The reader that is interested in the successive steps of the resampling can however skip it. The soft resampling scheme, which is based on importance sampling [LBD15], will be crucial in making the PF differentiable [ZMJ20, MKH20]. We might want, in general, to perform a subsampling of the particles, that is, we sample from the distribution in Eq. (B.2) not N but γN particles, with $0 < \gamma \leq 1$. We will later in the resampling routine propose $(1-\gamma)N$ new particles that will help us in representing the posterior better, so that we have in total after the resampling step N particles again. In this case the weights in Eq. (B.4) will be normalized as $w'_j \to \frac{w'_j}{\mathcal{C}}$, where $\mathcal{C}$ is such that

$$\frac{1}{\mathcal{C}}\sum_{j=1}^{\gamma N} w_j = \gamma , \tag{B.5}$$

By default we set $\gamma = 0.99$, that is, only 1% of the particles after the resampling are new.

Particle filters and importance sampling

In this section we review the core ideas underlying the functioning of a particle filter and the principle of importance sampling, as it is applied in our implementation of the soft resampling. Consider a distribution $P(\boldsymbol{\theta})$, from which we sample N particles $\boldsymbol{\theta}_j$ with $j = 1, \cdots, N$. Let us define an hypercube $\mathcal{C}$ of volume $\mathrm{d}\boldsymbol{\theta}$ centred around $\boldsymbol{\theta}$, and let us call $n(\boldsymbol{\theta}, d^N\boldsymbol{\theta})$ the number of particles in the said hypercube, i.e.

$$n(\boldsymbol{\theta}, \mathrm{d}\boldsymbol{\theta}) := \sum_{j=1}^{N} \chi_{\mathcal{C}}(\boldsymbol{\theta}_j) \tag{B.6}$$

with $\chi_{\mathcal{C}}$ being the characteristic function of the hypercube. We can write

$$\frac{1}{N} n(\boldsymbol{\theta}, d^N\boldsymbol{\theta}) \to P(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \quad \text{for } N \to \infty \,, \tag{B.7}$$

that is in the limit of large N the fraction of particles in the hypercube tends to the probability in such volume element. In a PF we associate to each particle $\boldsymbol{\theta}_i$ a weight w_i and we can define the total weight in the hypercube $\mathcal{C}$ as

$$P(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \simeq w(\boldsymbol{\theta}, d^N\boldsymbol{\theta}) := \sum_{j=1}^{N} w_j \chi_{\mathcal{C}}(\boldsymbol{\theta}_j) \,. \tag{B.8}$$

This total weight is the probability distribution actually represented by the PF. In the limit of large N, for a smooth distribution, we can consider the weight a function of the point $w(\boldsymbol{\theta})$, which varies smoothly in space and is approximatively constant in the hypercube $\mathcal{C}$. This leads us to write

$$P(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \simeq w(\boldsymbol{\theta}) \sum_{j=1}^{N} \chi_{\mathcal{C}}(\boldsymbol{\theta}_j) = w(\boldsymbol{\theta}) n(\boldsymbol{\theta}, \mathrm{d}\boldsymbol{\theta}) \,. \tag{B.9}$$

This means that the distribution represented by the particle filter is proportional to the product of the weights and the density of the particles. This is however not the only way to represent $P(\boldsymbol{\theta})$. Suppose that for whatever reason we sample the particles $\boldsymbol{\theta}_j$ from $Q(\boldsymbol{\theta})$, but that we actually want to represent the distribution $P(\boldsymbol{\theta})$. Then we can multiply the weights $w_j = 1/N$ of each particle $\boldsymbol{\theta}_j$ with the corrective factor $P(\boldsymbol{\theta}_j)/Q(\boldsymbol{\theta}_j)$, which remains

approximately constant inside the region $\mathcal{C}$, i.e.

$$w(\boldsymbol{\theta}, d^N\boldsymbol{\theta}) = \frac{1}{N}\sum_{j=1}^{N}\frac{P(\boldsymbol{\theta}_j)}{Q(\boldsymbol{\theta}_j)}\chi_{\mathcal{C}}(\boldsymbol{\theta}_j) \simeq \frac{P(\boldsymbol{\theta})}{Q(\boldsymbol{\theta})}\frac{1}{N}\sum_{j=1}^{N}\chi_{\mathcal{C}}(\boldsymbol{\theta}_j) = \frac{P(\boldsymbol{\theta})}{Q(\boldsymbol{\theta})}\frac{n(\boldsymbol{\theta}, d^N\boldsymbol{\theta})}{N} , \tag{B.10}$$

since now the particles are distributed spacially according to $Q(\boldsymbol{\theta})$ the density of particles will tend to $Q(\boldsymbol{\theta})$ for large N that, according to Eq. (B.7), gives $w(\boldsymbol{\theta}, d^N\boldsymbol{\theta}) \to Q(\boldsymbol{\theta})$, therefore we have $w(\boldsymbol{\theta}, d^N\boldsymbol{\theta}) \simeq P(\boldsymbol{\theta})$. In the case of soft resampling the distribution $Q(\boldsymbol{\theta})$ is constructed from $P(\boldsymbol{\theta})$ as

$$Q(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta} := \left[\alpha w(\boldsymbol{\theta}) + (1-\alpha)\frac{1}{N}\right] n(\boldsymbol{\theta}, \mathrm{d}\boldsymbol{\theta}) , \tag{B.11}$$

and the factor that multiplies the weights is $P(\boldsymbol{\theta})/Q(\boldsymbol{\theta}) = w(\boldsymbol{\theta})/\left[\alpha w(\boldsymbol{\theta}) + (1-\alpha)\frac{1}{N}\right]$. The factor used in Eq. (B.4) for the particle at $\boldsymbol{\theta}'_j$ contains $w_\phi(j)$, which is the weight at this point in the original distribution $P(\boldsymbol{\theta})$.

Gaussian perturbation

After the soft resampling we add a perturbation to the particles as proposed in [GFWC12], that is, we define

$$\boldsymbol{\theta}''_j := \beta\boldsymbol{\theta}'_j + (1-\beta)\widehat{\boldsymbol{\theta}}_t + \boldsymbol{\delta} , \tag{B.12}$$

where $\beta \in (0, 1]$, $\widehat{\boldsymbol{\theta}}_t$ is the mean of the posterior approximated in Eq. (5.8) and $\boldsymbol{\delta}$ is a random variable distributed according to

$$\boldsymbol{\delta} \sim \mathcal{N}(0, (1-\beta^2)\Sigma_t) . \tag{B.13}$$

With this expression we move the particles toward the mean of the posterior, which is our estimator for $\boldsymbol{\theta}$ and at the same time we lift the degeneracy of the $\boldsymbol{\theta}'_j$, that comes about because the particle $\boldsymbol{\theta}_j$ with high weights appear many times in the new particles ensemble. Were the degeneracy not removed, all these copies of the same particle wouldn't contribute much to improve resolution of the PF. This holds true unless they are perturbed, at which point they can encode the small scale behaviour of the posterior. Because of the perturbation in Eq. (B.12) the PF does not represent anymore the posterior $P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t)$ exactly. We now compute the probability distribution for $\boldsymbol{\theta}''_j$ after the perturbation step. The particles are distributed in the space according to the q_j weights in Eq. (B.2) and we call this distribution $Q(\boldsymbol{\theta}')$. Let us write Eq. (B.12) as $\boldsymbol{\theta}''_j = \beta\boldsymbol{\theta}'_j + \boldsymbol{\delta}'$ with

$$\boldsymbol{\delta}' \sim \mathcal{N}((1-\beta)\widehat{\boldsymbol{\theta}}_t, (1-\beta^2)\Sigma_t) , \tag{B.14}$$

being a perturbation with non-null mean value. Then the probability density for $\boldsymbol{\theta}''_j$ is the convolution of the probability of a particle being at position $\boldsymbol{\theta}'$ and the probability of the noise causing a displacement $\boldsymbol{\delta}' = \boldsymbol{\theta}'' - \beta\boldsymbol{\theta}'$, i.e.

$$\widetilde{Q}(\boldsymbol{\theta}'') = \int Q(\boldsymbol{\theta}') g_\beta(\boldsymbol{\theta}'' - \beta\boldsymbol{\theta}') \, \mathrm{d}\boldsymbol{\theta}' = \sum_{j=1}^{\gamma N} q_{\phi(j)} g_\beta(\boldsymbol{\theta}'' - \beta\boldsymbol{\theta}'_j) \; , \tag{B.15}$$

where g_β is the Gaussian probability density associated to $\boldsymbol{\delta}'$, i.e.

$$\begin{aligned} g_\beta(\boldsymbol{\theta}) := (2\pi)^{-\frac{\mathrm{d}}{2}} (1-\beta^2)^{-\frac{1}{2}} \det(\Sigma_t)^{-\frac{1}{2}} \\ \exp\left[-\frac{1}{2(1-\beta^2)} \left(\boldsymbol{\theta} - (1-\beta)\widehat{\boldsymbol{\theta}}_t \right)^\intercal \Sigma_t^{-1} \left(\boldsymbol{\theta} - (1-\beta)\widehat{\boldsymbol{\theta}}_t \right) \right] \; . \end{aligned} \tag{B.16}$$

In Eq. (B.15) we also substituted the integral with a summation being the probability $Q(\boldsymbol{\theta}')$ discrete. According to the principles of importance sampling the distribution represented by a PF is the product of the weights and the density of particles, which reads

$$\widetilde{P}(\boldsymbol{\theta}_j) \propto \frac{P(\boldsymbol{\theta})}{Q(\boldsymbol{\theta})} \widetilde{Q}(\boldsymbol{\theta}_j) \simeq P(\boldsymbol{\theta}) \; , \tag{B.17}$$

In principle we could correct the distribution for this perturbation by computing exactly Eq. (B.15) and accounting for it in the weights w'_j, in our implementation we don't do it however, since it would be very small anyway.

New particles proposal

We still need to produce $(1-\gamma)N$ new particles and we do it by extracting them from the Gaussian distribution with the same two first moments of the PF ensemble, i.e.

$$\boldsymbol{\theta}''_j \sim \mathcal{N}\left(\widehat{\boldsymbol{\theta}}, \Sigma \right) \; , \tag{B.18}$$

for $j = \gamma N, \cdots, N$. The mean and the covariance matrix are defined in Eq. (5.8) and Eq. (5.9) respectively. This is done again to increase the density of particles in the region of high probability, but it works properly only for unimodal distributions. The weights of these new particles are set to $w'_j = \frac{1}{N}$, so that their normalization is

$$\sum_{j=\gamma N}^{N} w'_j = 1 - \gamma \; . \tag{B.19}$$

This extra particles and weights are concatenated directly to $\{\boldsymbol{\theta}''_j\}_{j=1}^{\gamma N}$ and $\{w'_j\}_{j=1}^{\gamma N}$. We then rename the new weights and particles, i.e. $w'_j \to w_j$ and

$\boldsymbol{\theta}''_j \to \boldsymbol{\theta}_j$, and with that the resampling procedure is concluded. In doing the last step of proposing new particles we are mixing the distribution represented by the PF $P(\boldsymbol{\theta})$ as it comes out of the perturbation step in Eq. (B.17) with the distribution g_0 in Eq. (B.16). At the end the PF ensemble represents the distribution

$$P'(\boldsymbol{\theta}) = \gamma P(\boldsymbol{\theta}) + (1-\gamma) g_0(\boldsymbol{\theta}) \; . \tag{B.20}$$

Again we do not correct for this distortion, which could be done by modifying the weights properly.

Resampling of the batch

In order to compute the precision of the estimation, we need the results of many runs of the simulation, possibly executed in parallel on a GPU. In these circumstances the resampling is performed on all the instances of the estimation as soon as the condition Eq. (B.18) holds true for at least a fraction f of the estimations in the batch, which by default is set to $f = 0.98$. The premature resampling of an estimation run will have a quite strong detrimental effect on the goodness of the posterior represented by the PF, on the contrary a late resampling is much less probable to distort the distribution, this is the reason why we set f so close to one, that is, we want to limit as much as possible the number of simulations that are prematurely resampled. With the current implementation at each step either all the simulations is the batch are resampled or none. An improvement to the PF would be to resample selectively only those runs that are in need of resampling, and leave the other untouched until they satisfy Eq. (B.18), so that whatever number of runs could be resampled at each step. The complete resampling cycle, including the extraction and the new particle and the importance sampling is represented in Fig. B.1.

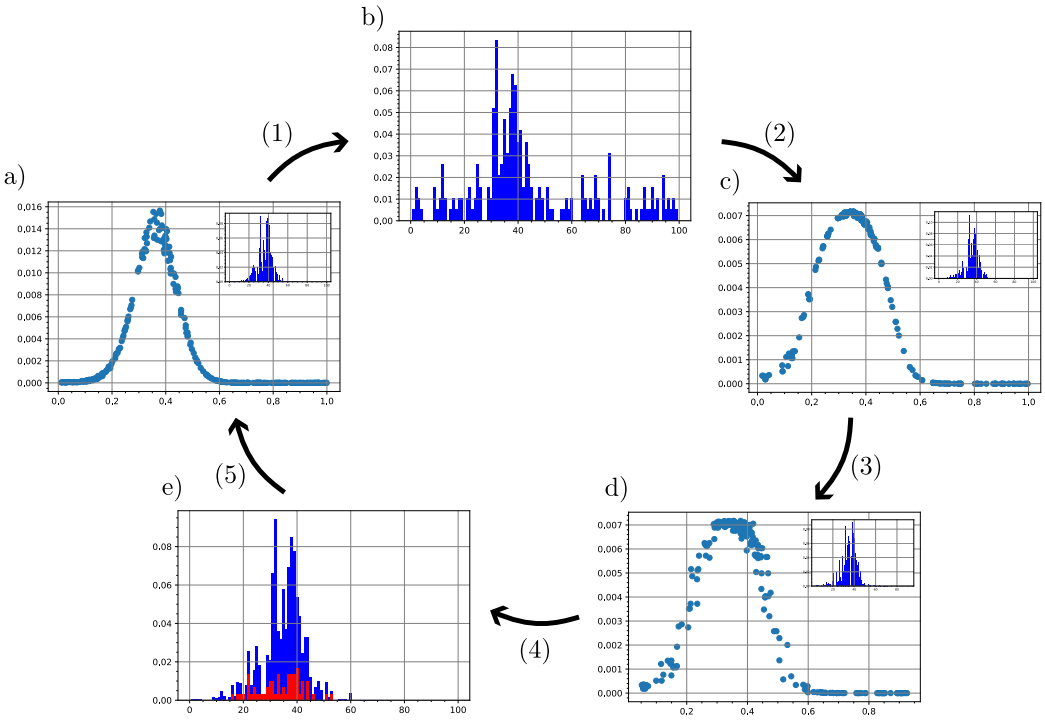


Figure B.1: The ensemble of the PF before the resampling is represented in a), the scatter plot are the points $(\boldsymbol{\theta}_j, w_j)$. This plot doesn't represent directly the posterior, because it doesn't take into account the density of particles. In all the plots, the inserted histogram is the actual posterior represented by the PF. Once the condition Eq. (B.1) is satisfy the first step of the resampling is executed, which is the transformation (1) and corresponds to sampling with repetitions from Eq. (B.2), the plot b) is the spatial density of particles after this action. The scatter plot c) is the distribution of the PF after the weights have been corrected according to Eq. (B.4) (transformation (2)). (3) is the application of the Gaussian noise in Eq. (B.12) and (4) is the sampling of the extra proposed particles in Eq. (B.18). Plot e) is the distribution represented by the PF when the resampling routine is complete, where in red the contribution of the new particles is highlighted. At last (5) is the repeated application of the Bayes rule Eq. (5.7), following some new measurement on the probe, that leads to the ensemble of the PF being again in need of resampling. To emphasise the effect of each transformation we have set $\alpha = 0.5, \beta = 0.9, \gamma = 0.8$. The total number of particles was $N = 10^3$ and the effective number of particles in a), that is before the resampling, was $N_{\text{eff}} = 93.8$. In the histograms the interval $[0, 1]$ of the scatter plots has been mapped to $[0, 100]$.

B.1.2 State particle filter

In this section we describe what happens when we are acting with weak (non-projective) measurements on the probe. In this case the probability to observe the outcome y_{t+1} at the step $t+1$ depends on all the string of previous outcomes and controls, that is on the whole trajectory $\tau := (\boldsymbol{x}_t, \boldsymbol{y}_t)$, as well as on the current control x_{t+1}. This means we must substitute $p(y_{t+1}|x_{t+1}, vtheta)$ with $p(y_{t+1}|\boldsymbol{\theta}, \boldsymbol{x}_{t+1}, \boldsymbol{y}_t)$ in Eq. (5.7). Since we avoid the reinitialization of the probe, its state depends on all the evolution history. With this change in the outcome probability all the formulas of the previous section remain valid. To compute $p(y_{t+1}|\boldsymbol{\theta}, \boldsymbol{x}_{t+1}, \boldsymbol{y}_t)$, we need to keep track of the state of the probe. In order to do so we introduce the state particle filter. In this data structure we save for each particle $\boldsymbol{\theta}_j$ the state of probe had the system evolved under the action of $\mathcal{E}_{\boldsymbol{\theta}_j,x}$, with the controls and the outcomes being the ones actually applied/observed in the evolution, we indicate this state with $\rho_{\boldsymbol{\theta}_j,\tau_t}$. To this state we associate the weight w_j of the particle $\boldsymbol{\theta}_j$. The state particle filter represents the posterior probability distribution for the state of the probe conditioned on the trajectory τ_t. The expression for $\rho_{\boldsymbol{\theta}_j,\tau_t}$ reads

$$\rho_{\boldsymbol{\theta}_j,\tau_t} = \mathcal{M}^{x_t}_{y_t} \circ \mathcal{E}_{\boldsymbol{\theta}_j,x_t} \circ \mathcal{M}^{x_{t-1}}_{y_{t-1}} \circ \mathcal{E}_{\boldsymbol{\theta}_j,x_{t-1}} \circ \cdots \circ \mathcal{M}^{x_1}_{y_1} \circ \mathcal{E}_{\boldsymbol{\theta}_j,x_1}(\rho) \ , \tag{B.21}$$

where

$$\mathcal{M}^{x_t}_{y_t}(\rho) := \frac{M^{x_t}_{y_t} \rho M^{x_t\dagger}_{y_t}}{\mathrm{tr}\left[M^{x_t}_{y_t} \rho M^{x_t\dagger}_{y_t}\right]} \ , \tag{B.22}$$

is the backreaction of the measurement on the state of the probe. The estimator for the probe state at the step t is

$$\widehat{\rho}_{\tau_t} := \sum_{j=1}^{N} w_j \rho_{\boldsymbol{\theta}_j,\tau_t} \ , \tag{B.23}$$

that is, the mean of the state on the posterior distribution for the parameters. The estimator $\widehat{\rho}_{\tau_t}$ can then be fed to the agent, to contribute to the computation of the next control. When the resampling is performed on the PF ensemble we get a new set of particles $\boldsymbol{\theta}'_j$ and their corresponding states must be also updated. This means we have to keep track of the vectors $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$ in the simulations and recompute the evolution of the whole state particle filter from the beginning, so that we get $\rho_{\boldsymbol{\theta}'_j,\tau_t}$. From the computational point of view, the fact that we need these rather memory intensive structures of the PF and the state PF tells us that the optimization loop presented here can be applied only to rather small and simple quantum sensors.

B.1.3 Multimodal posterior distributions

The resampling procedure presented in the previous section has some limitation in dealing with multimodal distributions. In this case the mean of the posterior may lay in a region of relatively low probability between two peaks and the accumulation of particles in this region after a resampling would be detrimental to the precision of the estimation. From its own design it would be difficult to modify the PF so that it accounts for multiple maxima. The informations that we can easily extract from the PF are its moments and from them the actual positions of the maxima are not straightforward to obtain. Multimodal posterior distributions are however common in quantum metrology. For example in multiphase estimation, like the measurement of the hyperfine interaction in NV-^{13}C. To promote the preservation of secondary features in the posterior distribution we can use multiple particle filter at once. In this situation a set of PFs, with different priors, are updated in parallel and only together represent the full Bayesian posterior. To reduce the memory requirement of such approach we could consider simple Gaussian distributions instead of full PFs. We start by approximating the prior distribution $\pi(\boldsymbol{\theta})$ with as a sum of L Gaussians:

$$\pi(\boldsymbol{\theta}) \simeq \sum_{l=1}^{L} w_l \mathcal{N}(\boldsymbol{\mu}_l, \boldsymbol{\sigma}_l) \ . \tag{B.24}$$

If the parameters $\boldsymbol{\mu}_l, \boldsymbol{\sigma}_l$ are fixed then the Bayesian update step can be done by solving a linear regression problem to find the best new values for $\{w_l\}_{l=1}^{L}$ that represent the posterior. In this way the PF has however a limited resolution, determined by the initial Gaussian. If we also let $\boldsymbol{\mu}_l, \boldsymbol{\sigma}_l$ change during the Bayesian update step, then we solve the problem of having limited resolution, but we now have to deal with a non-linear regression problem.

B.2 Differentiability of the particle filter

In this section we discuss what happens when the resampling routine of the particle filter is switched on, and, in particular, what we need to do to assure that the gradient produced by the automatic differentiation is correct.

B.2.1 Differentiable PF through reparametrization and soft resampling

Differentiability of the soft resampling

As seen in Section B.3.4, the gradient can't be propagated through randomly extracted variables, therefore when the categorical resampling is executed, the particles $\boldsymbol{\theta}'_j$ in Eq. (B.3) don't have any connection with the controls, i.e.

$$\frac{\mathrm{d}\boldsymbol{\theta}'_j}{\mathrm{d}\boldsymbol{\lambda}} = 0 \ . \tag{B.25}$$

Similarly, the weights are reinitialized and lose every dependence on the history of the estimation. At the moment of taking the gradient we won't be able to account for anything that happened before the last resampling. This means that the training routine optimizes the agent only for the later steps, although what has been learnt in this context may be useful also for the earlier measurements. The soft resampling with $\alpha < 1$, introduced in Appendix B.1, is able to partially remove this obstacle. With this trick the dependence on $\boldsymbol{\lambda}$ is passed from the old weights to the new ones through Eq. (B.4). However, the gradient doesn't backpropagate entirely but it is attenuated by the factor $1 - \alpha$. The price to pay for propagating the gradient is that the N particles are not all fully effective for the resampling, instead only a fraction α of them participate to it. As discussed in Section B.3.4, since we are extracting stochastic variables from the distribution in Eq. (B.2), we should also add the corresponding log-likelihood terms $\sum_j \log q_{\phi(j)}$ to the loss. However adding so many terms would increase too much the variance of the gradient. Either we don't account for these log-likelihoods and we accept the gradient to have a bias, or we use the correction introduced in [SW21], that prescribes to substitute the definition of the new weights w'_j with an appropriate surrogate expressions. Introducing this correction is the default behaviour of our code but it can be applied only if the loss $\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})$ is of a certain form. It can be proved that for the MSE defined in Eq. (5.15) this gives the correct gradient. See Section B.2.2 for a complete discussion.

Differentiability of the perturbation

The next transformation on the particles is the perturbation of Eq. (B.12). Again we would be unable to propagate the gradient through the perturbation $\boldsymbol{\delta}'$, if we extracted it directly from the Gaussian $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$, with $\boldsymbol{\mu} = (1 - \beta)\widehat{\boldsymbol{\theta}}_t$ and $\Sigma = (1 - \beta^2)\Sigma_t$. For this reason we apply the reparametrization trick and write

$$\boldsymbol{\delta}'(\boldsymbol{y}_t, \boldsymbol{\lambda}) = \Sigma(\boldsymbol{y}_t, \boldsymbol{\lambda})\boldsymbol{u} + \boldsymbol{\mu}(\boldsymbol{y}_t, \boldsymbol{\lambda}) \ , \tag{B.26}$$

where $\boldsymbol{u}$ is extracted from the multivariate standard Gaussian $\boldsymbol{u} \sim \mathcal{N}(0, \mathbb{1})$. The perturbation is now a differentiable functions of $\boldsymbol{\lambda}$. For the extraction of $\boldsymbol{u}$ we do not need to add a corresponding log-likelihood term, as discussed in Section B.3.4, because its probability density function doesn't depend on $\boldsymbol{\lambda}$.

Differentiability of the proposed particles

For the last step of the resampling, which consists in proposing new particles $\boldsymbol{\theta}''_j$, extracted from $\mathcal{N}(\widehat{\boldsymbol{\theta}}_t, \Sigma_t)$, we again exploit the reparametrization trick and write

$$\boldsymbol{\theta}''_j(\boldsymbol{y}_t, \boldsymbol{\lambda}) = \Sigma_t(\boldsymbol{y}_t, \boldsymbol{\lambda})\boldsymbol{u}_j + \widehat{\boldsymbol{\theta}}_t(\boldsymbol{y}_t, \boldsymbol{\lambda}) \,, \tag{B.27}$$

which is again differentiable and doesn't require a log-likelihood term.

Differentiability of the state particle filter

Regarding the differentiability of the state particle filter discussed in Section B.1.2, we observe that its elements are functions of $\boldsymbol{\lambda}$ through the trajectory $\tau_t(\boldsymbol{\lambda}) = (\boldsymbol{x}_t(\boldsymbol{y}_t, \boldsymbol{\lambda}), \boldsymbol{y}_t)$:

$$\rho_j(\boldsymbol{\lambda}) := \rho_{\boldsymbol{\theta}_j, \tau_t(\boldsymbol{\lambda})} \,. \tag{B.28}$$

Under the assumption that the encoding and the measurements appearing in Eq. (B.21) are differentiable we can propagate the gradient through the evolution of the probe in Eq. (B.21) . When the particles are resampled and the new states are computed, the dependence of the new state $\rho_j(\boldsymbol{\lambda})$ on the old weights of the PF, and therefore on $\boldsymbol{\lambda}$, persists through the measurement backreaction operators $\mathcal{M}^{y_t}_{x_t(\boldsymbol{\lambda})}$. A new dependence on $\boldsymbol{\lambda}$ appears in the evolution map $\mathcal{E}_{\boldsymbol{\theta}_j(\boldsymbol{\lambda}), x_t(\boldsymbol{\lambda})}$, coming from the new particles $\boldsymbol{\theta}_j(\boldsymbol{\lambda})$, so that we can write

$$\rho'_j(\boldsymbol{\lambda}) := \rho_{\boldsymbol{\theta}_j(\boldsymbol{\lambda}), \tau_t(\boldsymbol{\lambda})} \,. \tag{B.29}$$

B.2.2 Differentiable PF through the correction of Ścibior and Wood

In [SW21] a correction was introduced to make the resampling procedure in a PF differentiable. This can be implemented in place of the soft resampling, or alongside with it. The default behaviour of our software is to perform the soft resampling with $\alpha = 0.5$ alongside the Ścibor and Wood correction. With this choice only half of the gradient is backpropagated through soft

resampling, the other half is done by the Ścibor and Wood correction. The former prescribes to modify the normalized weights w'_j of Eq. (B.4) to

$$\widetilde{w}'_j \leftarrow w'_j \frac{q_{\phi(j)}}{\mathrm{sg}\left[q_{\phi(j)}\right]} \,, \tag{B.30}$$

where the meaning of the symbols is that of Section B.1.1. In this formula we are using the stop gradient operator $\mathrm{sg}\,[\cdot]$, which is an instruction that tells the automatic differentiation frameworks not to compute the derivatives of the expression inside the operator. This correction has no effects in the forward pass, but produces additional gradient terms in the backward pass. We see in this section, that for a MSE loss the extra terms in the gradient appearing because of this surrogate expression are exactly the log-likelihoods that we would have to insert following the conclusions of Section B.3.4, although this observation can't be extended to a generic loss. Let us start from the expression for the MSE, when one and only one resampling is performed in the whole experiment, at step t, i.e.

$$\Delta^2\widehat{\boldsymbol{\theta}} = \int \ell(\widehat{\boldsymbol{\theta}},\boldsymbol{\theta}) P(\tau_{t+1:M-1}|\boldsymbol{\theta}'_j,\boldsymbol{\theta}) \left(\prod_{j=1}^{N} P(\boldsymbol{\theta}'_j|\tau_{0:t})\right) P(\tau_{0:t}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})\,\mathrm{d}\tau_{M-1} \left(\prod_{j=1}^{N}\mathrm{d}\boldsymbol{\theta}'_j\right)\mathrm{d}\boldsymbol{\theta}\,. \tag{B.31}$$

We assume for clarity that the perturbation and the extraction of the extra particles, in Eq. (B.26) and Eq. (B.27) respectively, are turned off, i.e. $\beta = \gamma = 1$. The object $\tau_{\alpha:\beta}$ with α, β integers in $[0, M-1]$ is the trajectory between the steps α and β (extrema included), i.e. $\tau_{\alpha:\beta} = (\boldsymbol{x}_{\alpha:\beta}, \boldsymbol{y}_{\alpha:\beta})$ with $\boldsymbol{x}_{\alpha:\beta} = (x_\alpha, x_{\alpha+1}, \ldots, x_\beta)$ and $\boldsymbol{y}_{\alpha:\beta} = (y_\alpha, y_{\alpha+1}, \ldots, y_\beta)$. Reading Eq. (B.31) from right to left we encounter the probability densities for all the random variable extractions in chronological order. First the extraction of the true values $\boldsymbol{\theta}$ for the simulation instance, then the trajectory up to the resampling point, then the extraction of the new particles $\boldsymbol{\theta}'_j$, and finally the measurements after the resampling, i.e. the trajectory after the t-th step until the end. This last probability depends also on the values of the new particles, through the posterior distribution momenta, that are passed to the agent that decides the next control. We now insert the expression for $\ell(\widehat{\boldsymbol{\theta}},\boldsymbol{\theta})$ found in Eq. (5.15) in Eq. (B.31). We postpone the computation of the trace to the end and expand the error matrix for the estimator in Eq. (5.8), i.e.

$$(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta})(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta})^\intercal = \sum_{i,j=1}^{N} w'_i w'_j \boldsymbol{\theta}'_i \boldsymbol{\theta}'^\intercal_j - \boldsymbol{\theta}\left(\sum_{j=1}^{N} w'_j \boldsymbol{\theta}'^\intercal_j\right) - \left(\sum_{j=1}^{N} w'_j \boldsymbol{\theta}'_j\right)\boldsymbol{\theta}^\intercal + \boldsymbol{\theta}\boldsymbol{\theta}^\intercal \,. \tag{B.32}$$

Each term in the first summation gives a contribution to $\Delta^2\widehat{\boldsymbol{\theta}}$ equal to

$$\int w'_i w'_j \boldsymbol{\theta}'_i \boldsymbol{\theta}'^{\intercal}_j P(\tau_{t+1:M-1}|\boldsymbol{\theta}) P(\boldsymbol{\theta}'_i|\tau_{0:t}) P(\boldsymbol{\theta}'_j|\tau_{0:t}) P(\tau_{0:t}|\boldsymbol{\theta}) \pi(\boldsymbol{\theta}) \,\mathrm{d}\tau_{M-1} \,\mathrm{d}\boldsymbol{\theta}'_i \,\mathrm{d}\boldsymbol{\theta}'_j \,\mathrm{d}\boldsymbol{\theta} \;, \tag{B.33}$$

where we neglect all the integrals on the variables $\boldsymbol{\theta}'_\alpha$ with index $\alpha \neq i, j$, because they do not appear in the integrand. The gradient of this term with respect to $\boldsymbol{\lambda}$ gives rise to the usual likelihood terms for the measurement plus the following extra terms coming from the resampling:

$$\sum_{i,j}^{N} w'_i w'_j \boldsymbol{\theta}'_i \boldsymbol{\theta}'^{\intercal}_j \left(\frac{\mathrm{d} \log P(\boldsymbol{\theta}'_i|\tau_{0:t})}{\mathrm{d}\boldsymbol{\lambda}} + \frac{\mathrm{d} \log P(\boldsymbol{\theta}'_j|\tau_{0:t})}{\mathrm{d}\boldsymbol{\lambda}} \right) \;, \tag{B.34}$$

for $i, j = 1, \ldots, N$. Similarly the linear terms in Eq. (B.32) give the following likelihood terms:

$$-\sum_{j=1}^{N} w'_j \left(\boldsymbol{\theta}'_j \boldsymbol{\theta}^{\intercal} + \boldsymbol{\theta} \boldsymbol{\theta}'^{\intercal}_j \right) \frac{\mathrm{d} \log P(\boldsymbol{\theta}'_j|\tau_{0:t})}{\mathrm{d}\boldsymbol{\lambda}} \;, \tag{B.35}$$

plus the same terms with $\boldsymbol{\theta}'^{\intercal}_j$. There is no likelihood terms associated to the constant $\boldsymbol{\theta}\boldsymbol{\theta}^{\intercal}$ in Eq. (B.32). Now we shall see that deriving the surrogate expression gives the same terms in the gradient. Let us write the total derivative of the error matrix:

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})^{\intercal} &= \frac{\mathrm{d}\widehat{\boldsymbol{\theta}}}{\mathrm{d}\boldsymbol{\lambda}} (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})^{\intercal} + (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \frac{d\widehat{\boldsymbol{\theta}}^{\intercal}}{\mathrm{d}\boldsymbol{\lambda}} \\ &= \left(\sum_{i=1}^{N} \frac{\mathrm{d}\widetilde{w}'_i}{\mathrm{d}\boldsymbol{\lambda}} \boldsymbol{\theta}'_i \right) \left(\sum_{j=1}^{N} w'_j \boldsymbol{\theta}'_j - \boldsymbol{\theta} \right)^{\intercal} \\ &+ \left(\sum_{i=1}^{N} w'_i \boldsymbol{\theta}'_i - \boldsymbol{\theta} \right) \left(\sum_{j=1}^{N} \frac{\mathrm{d}\widetilde{w}'_j}{\mathrm{d}\boldsymbol{\lambda}} \boldsymbol{\theta}'^{\intercal}_j \right) \;. \end{aligned}$$

Where the derivative doesn't act the weights $\widetilde{w}'_j$ become w'_j. From the definition of $\widetilde{w}'_j$ in Eq. (B.30) we compute the derivative of the surrogate expression, that is

$$\frac{\mathrm{d}\widetilde{w}'_j}{\mathrm{d}\boldsymbol{\lambda}} = \frac{dw'_j}{\mathrm{d}\boldsymbol{\lambda}} + w'_j \frac{\mathrm{d} \log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}} \;. \tag{B.36}$$

We know keep track only of the extra likelihood terms coming from the surrogate part of the weights $\widetilde{w}'_j$ and organize the terms according to the order in $\boldsymbol{\theta}'_j$. We have the second order terms:

$$\sum_{i,j}^{N} w'_i w'_j \boldsymbol{\theta}'_i \boldsymbol{\theta}'^{\intercal}_j \left(\frac{\mathrm{d} \log q_{\phi(i)}}{\mathrm{d}\boldsymbol{\lambda}} + \frac{\mathrm{d} \log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}} \right) \;, \tag{B.37}$$

and the first order ones:

$$\sum_{j=1}^{N} w'_j \left(\boldsymbol{\theta}'_j \boldsymbol{\theta}^\intercal + \boldsymbol{\theta}\boldsymbol{\theta}'^\intercal_j\right) \frac{\mathrm{d}\log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}} \ , \tag{B.38}$$

which correspond respectively to Eq. (B.34) and Eq. (B.35), once we realize that $P(\boldsymbol{\theta}'_j|\tau_{0:t}) = q_{\phi(j)}$. One of the advantages of this approach is the reduce variance of the gradient estimator, which would explode, where we to insert all the likelihood terms for the new particle extractions at the end of the estimation. The correction, however, doesn't produce always the correct gradient for the loss, but only when $\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})$ is a polynomial function of the weights $\widetilde{w}'_j$. Consider the estimation of single parameter $\theta \in [0, 2\pi)$, we might want to use a loss functions $l(\widehat{\theta}, \theta)$ that respect the circular nature of the parameter, like

$$l(\widehat{\theta}, \theta) := \sin(\widehat{\theta} - \theta)^2 \ . \tag{B.39}$$

The gradient with respect to $\boldsymbol{\lambda}$, when the correction is implemented, is

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} l(\widehat{\theta}, \theta) &= 2\sin(\widehat{\theta} - \theta)\cos(\widehat{\theta} - \theta)\frac{\mathrm{d}\widehat{\theta}}{\mathrm{d}\boldsymbol{\lambda}} && \text{(B.40)} \\
&= \sin(2\widehat{\theta} - 2\theta)\sum_{j=1}^{N} \frac{\mathrm{d}\widetilde{w}'_j}{\mathrm{d}\boldsymbol{\lambda}}\theta'_j && \text{(B.41)} \\
&= \sin(2\widehat{\theta} - 2\theta)\sum_{j=1}^{N} \left(\frac{dw'_j}{\mathrm{d}\boldsymbol{\lambda}}\theta'_j + w'_j\theta'_j\frac{\mathrm{d}\log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}}\right) \ , && \text{(B.42)}
\end{aligned}$$

so that the likelihood term in the gradient is

$$\sin(2\widehat{\theta} - 2\theta)\sum_{j=1}^{N} w'_j\theta'_j\frac{\mathrm{d}\log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}} \ , \tag{B.43}$$

while it should be

$$\sin(\widehat{\theta} - \theta)^2\sum_{j=1}^{N} \frac{\mathrm{d}\log q_{\phi(j)}}{\mathrm{d}\boldsymbol{\lambda}} \ . \tag{B.44}$$

Another example where the correction fails in the loss of Eq. (5.16). Let us take $\widehat{\boldsymbol{\theta}}$ to be the maximum likelihood estimator, then, since a small perturbation in the posterior distribution won't change it we have

$$\frac{\mathrm{d}\widehat{\boldsymbol{\theta}}}{\mathrm{d}\boldsymbol{\lambda}} = \frac{\mathrm{d}\widehat{\boldsymbol{\theta}}}{\mathrm{d}w_j}\frac{\mathrm{d}w_j}{\mathrm{d}\lambda} = 0 \ , \tag{B.45}$$

therefore the correction is useless to backpropagate the gradient through the resampling step and we must rely only on the importance sampling. Incidentally we notice that the loss for Eq. (5.16) is a pure-likelihood expression, analogous to the loss in regular Policy Gradient RL.

B.3 Computation and differentiation of the loss function

As in most optimization problems, the trainable variables of the agent are updated with a version of the stochastic gradient descent. In this section, we define the loss function for this training, compute its gradient, and comment on the computational resources required by the training.

B.3.1 Definition of the loss function

The two scalar losses that we used in this work are the MSE, defined in Eq. (5.15), used for continuous parameters, and the discrimination loss of Eq. (5.16), used for discrete parameters, that converges to the error probability when averaged. If the parameter to be estimated is a phase we might want to take as loss the circular variance [Hol11]. In the following section we adopt the symbol $\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})$ for the loss and keep the discussion completely general. We mention that this analysis would apply also to a more general class of losses, being functions of the of the PF ensemble, i.e. $\ell(\mathfrak{p}, \boldsymbol{\theta})$, provided they are well-behaved as functions. The expected value of the loss on the trajectory is

$$\Delta^2\widehat{\boldsymbol{\theta}}_\tau := \int \ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta}) P(\widehat{\boldsymbol{\theta}}|\tau_{M-1}, \boldsymbol{\theta})\, \mathrm{d}\widehat{\boldsymbol{\theta}}\ , \tag{B.46}$$

with $\tau_{M-1} := (\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1})$ indicating the complete trajectory. This definition presumes a stochastic dependence of the estimator $\widehat{\boldsymbol{\theta}}$ computed from the PF on the outcomes and the controls of the measurements, collectively denominated τ_{M-1}. This is codified by the probability density $P(\widehat{\boldsymbol{\theta}}|\tau_{M-1}, \boldsymbol{\theta})$. This stochasticity can be due to the resampling routine or, in general, to the construction of the estimator $\widehat{\boldsymbol{\theta}}$, which could entail the sampling from a distribution, which is however never the case in our examples. The quantity $\Delta^2\widehat{\boldsymbol{\theta}}_\tau$ refers to a single trajectory of the PF, the one indicated with τ_{M-1}. We wish however to consider the average of the MSE over all the possible trajectories τ_{M-1} weighted appropriately. The expectation value over τ_{M-1} is expressed by the following operator

$$\mathbb{E}_\tau\left[\cdot\right] := \int \cdot\, P(\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1}|\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{x}_{M-1}\, \mathrm{d}\boldsymbol{y}_{M-1}\ , \tag{B.47}$$

which applied to Eq. (B.46) gives

$$\mathbb{E}_\tau\left[\Delta^2\widehat{\boldsymbol{\theta}}_\tau\right] = \int \ell(\widehat{\boldsymbol{\theta}},\boldsymbol{\theta})P(\widehat{\boldsymbol{\theta}}|\boldsymbol{\theta})\,\mathrm{d}\widehat{\boldsymbol{\theta}}\,, \tag{B.48}$$

where we have defined

$$P(\widehat{\boldsymbol{\theta}}|\boldsymbol{\theta}) := \int P(\widehat{\boldsymbol{\theta}}|\tau_{M-1},\boldsymbol{\theta})P(\tau_{M-1}|\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{x}_{M-1}\,\mathrm{d}\boldsymbol{y}_{M-1}\,. \tag{B.49}$$

We also want to take the expectation value of $\widehat{\boldsymbol{\theta}}$ on the prior $\pi(\boldsymbol{\theta})$ through the operator

$$\mathbb{E}_{\boldsymbol{\theta}}\left[\cdot\right] := \int \cdot\,\pi(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}\,, \tag{B.50}$$

which applied to $\mathbb{E}_\tau\left[\Delta^2\widehat{\boldsymbol{\theta}}_\tau\right]$ gives the figure of merit for the error

$$\Delta^2\widehat{\boldsymbol{\theta}} := \mathbb{E}_{\boldsymbol{\theta}}\left[\mathbb{E}_\tau\left[\Delta^2\widehat{\boldsymbol{\theta}}_\tau\right]\right] = \int \ell(\widehat{\boldsymbol{\theta}},\boldsymbol{\theta})P(\widehat{\boldsymbol{\theta}})\,\mathrm{d}\widehat{\boldsymbol{\theta}}\,, \tag{B.51}$$

with

$$P(\widehat{\boldsymbol{\theta}}) := \int P(\widehat{\boldsymbol{\theta}}|\tau_{M-1})P(\tau_{M-1}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{x}_{M-1}\,\mathrm{d}\boldsymbol{y}_{M-1}\,\mathrm{d}\boldsymbol{\theta}\,. \tag{B.52}$$

This is the probability density for the final estimator $\widehat{\boldsymbol{\theta}}$, given that the true value $\boldsymbol{\theta}$ is extracted from the prior $\pi(\boldsymbol{\theta})$ at the beginning and we average over the trajectory τ_{M-1} that is stochastically generated in the simulation, through the actions of the agent and the measurements. The expression in Eq. (B.51) suggests us a straightforward way to approximate the error from the numerical simulation (*à la* Monte Carlo), i.e.

$$\Delta^2\widehat{\boldsymbol{\theta}} \simeq \frac{1}{B}\sum_{k=1}^{B}\ell(\widehat{\boldsymbol{\theta}}_k,\boldsymbol{\theta}_k)\,, \tag{B.53}$$

where $\boldsymbol{\theta}_k$ is the true value of the parameters in the k-th simulation and $\widehat{\boldsymbol{\theta}}_k$ is the corresponding final estimator. By carrying out the complete estimation in a batch of B simulated experiments, with each $\boldsymbol{\theta}_k$ extracted from $\pi(\boldsymbol{\theta})$, we are effectively sampling $\widehat{\boldsymbol{\theta}}$ from $P(\widehat{\boldsymbol{\theta}})$ so that by the law of large number we can approximate the expectation value of the loss function in Eq. (B.51) with the empirical mean on the batch. Each simulation in the batch follows its particular trajectory, which will be different from the ones of the other

simulations, because the randomly extracted measurement outcomes are different. Notice that in distinction with the notation of the previous sections the subscript in Eq. (B.53) doesn't refer to the step of the measurement cycle, but to the index of the simulation in the batch: the estimators $\widehat{\boldsymbol{\theta}}_k$ are always evaluated at the last step $t = M - 1$. We call B the *batchsize* of the simulation. The right-hand side of Eq. (B.53) will be the loss to be minimized by the training procedure. A natural question that arises here, is why aren't we using the covariance matrix as estimated from the PF in the computation of the MSE? The answer is that the PF may be imprecise for the evaluation of the variance, in particular, it tends to underestimate it, because some tails of the distribution $P(\widehat{\boldsymbol{\theta}}|\tau_{M-1})$ may not be very well represented. We prefer to estimate the MSE empirically from the sampled $\widehat{\boldsymbol{\theta}}_k$, extracted from the true distribution $P(\widehat{\boldsymbol{\theta}})$, in order to avoid biases. The loss of Eq. (B.53) is the closest it can be to the precision we would observe in an experiment.

B.3.2 Definition of the loss for limited resources

In the previous paragraph we have implicitly assumed that the stopping condition of the estimation was based on the number of measurement M, i.e. we had a fixed number of measurement in each instance of the estimation. If, however, the resources are not simply related to the number of measurement steps, since each estimation in the batch follows its own trajectory, we may have different termination times, which correspond to the sensor employing a different number of measurement steps to consume all the available resources. In this section we introduce the notation $\widehat{\boldsymbol{\theta}}_{k,t}$, where the first subscript k is the index in the batch, and the second t is the measurement step. Whatever the nature of the resource chosen, to avoid having infinite loops we always fix a maximum number of measurement steps M in the simulation, that should to be much larger than the expected number of iterations before the resources run out. At each step only the PF ensemble of those estimations which haven't terminated yet are updated with the Bayes rule, all the others, which have already consumed the available amount of resources, remain "freezed", since no measurement is performed and therefore no update is applied. Nevertheless all the quantities computed from the PF ensemble, e.g. $\widehat{\boldsymbol{\theta}}_{k,t}$ and $\Sigma_{k,t}$ are defined potentially for all the estimation steps $t = 0, \cdots, M - 1$. To put it simply if $t_k^\star$ was the index of the last measurement for the k-th estimation in the batch before it running out of resources, then $\widehat{\boldsymbol{\theta}}_{k,t} = \widehat{\boldsymbol{\theta}}_{t_k^\star,k}$, $\Sigma_{k,t} = \Sigma_{t_k^\star,k}$ for $t \geq t_k^\star$. In general, the PF ensemble remains the same if no new measurement outcomes are incorporated, i.e. $\mathfrak{p}_{k,t} = \mathfrak{p}_{t_k^\star,k}$ for $t \geq t_k^\star$. The simplest stopping condition for the measurement cycle is now that all the B estimations in the

batch have concluded, but to reduce the simulation time we only ask for at least a fraction $\nu = 0.98$ of estimations to have terminated. These would exclude those simulations that are taking too long to terminate. We define M' the realized number of iterations in the measurement loop determined by this condition, so that the loss in Eq. (B.53) becomes

$$\Delta^2\widehat{\boldsymbol{\theta}} \simeq \frac{1}{B'}\sum_{k=1}^{B'} \ell(\widehat{\boldsymbol{\theta}}_{k,M'}, \boldsymbol{\theta}_k) \,, \tag{B.54}$$

where the summation is taken only on those $B' = \lceil \nu B \rceil$ estimations in the batch that have terminated.

B.3.3 Dependence of the loss on the trainable variables

We go on by deriving from Eq. (B.51) an expression for the MSE, that is more directly related to the quantities simulated, under the hypotheses that the resampling has been turned off, i.e. $r_t = 0$, and that the computation of $\widehat{\boldsymbol{\theta}}$ from the ensemble of the PF doesn't require any stochastic operation. These are working hypotheses, which will allow to make useful observations and generalizations, whose domain of applications is however not limited by the said hypotheses. We begin observing that the controls $\boldsymbol{x}_{M-1}$ produced by the agent are deterministic functions of the ensemble of the PF, for example through the mean and the covariance matrix. Therefore, the weights of the PF are in turn deterministic functions of the measurement outcomes, as they are computed with Eq. (5.7), so that we can write the controls $\boldsymbol{x}_t$ and the estimator $\widehat{\boldsymbol{\theta}}_t$ at step t as

$$\boldsymbol{x}_t = g_1(\boldsymbol{y}_{t-1}, \boldsymbol{\lambda}) \quad \text{and} \quad \widehat{\boldsymbol{\theta}}_t = g_2(\boldsymbol{y}_t, \boldsymbol{\lambda}) \,, \tag{B.55}$$

for two appropriate functions g_1 and g_2. Beside the outcomes both the controls and the estimators depend on the trainable variables of the agent, indicated with $\boldsymbol{\lambda}$, for the aforementioned reasons. Under these hypotheses the probabilities appearing in Eq. (B.52) can be rewritten as

$$P(\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1}|\boldsymbol{\theta}) = \delta(\boldsymbol{x}_{M-1} - g_1(\boldsymbol{y}_{M-2}, \boldsymbol{\lambda}))p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \,, \tag{B.56}$$

$$P(\widehat{\boldsymbol{\theta}}|\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1}) = \delta(\widehat{\boldsymbol{\theta}} - g_2(\boldsymbol{y}_{M-1}, \boldsymbol{\lambda})) \,. \tag{B.57}$$

Solving the integrals in $\mathrm{d}\boldsymbol{x}_{M-1}$ and in $\mathrm{d}\widehat{\boldsymbol{\theta}}$ in Eq. (B.51), we get the following expression for the MSE

$$\Delta^2\widehat{\boldsymbol{\theta}} = \int \ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1}, \boldsymbol{\lambda}), \boldsymbol{\theta}) p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \pi(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{y}_{M-1} \, \mathrm{d}\boldsymbol{\theta} \,. \tag{B.58}$$

This is an expectation value on the probability distribution of the tuple of outcomes $\boldsymbol{y}_{M-1}$. We introduce $\omega := (\boldsymbol{y}_{M-1}, \boldsymbol{\theta})$ and redefine the loss for the next sections as

$$\ell(\omega, \boldsymbol{\lambda}) := \ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1}, \boldsymbol{\lambda}), \boldsymbol{\lambda}) \ . \tag{B.59}$$

The object ω contains all the variables that depend on the specific instance of the simulation so that the empirical approximation of $\Delta^2\widehat{\boldsymbol{\theta}}$ from Eq. (B.58) is

$$\mathcal{L}(\boldsymbol{\lambda}) := \frac{1}{B}\sum_{k=1}^{B} \ell(\omega_k, \boldsymbol{\lambda}) \ , \tag{B.60}$$

with $\omega_k := (\boldsymbol{y}_{k,M-1}, \boldsymbol{\theta}_k)$. The true values $\boldsymbol{\theta}_k$ are sampled from $\pi(\boldsymbol{\theta})$ at the beginning of the run. In case the agent is a NN the trainable variables are the weights and the biases, while for the non-adaptive strategy the variables are directly the tuple of all the controls, i.e. $\boldsymbol{\lambda} = (x_1, x_2, \ldots, x_{M-1})$. The average loss in Eq. (B.60) will be also named the scalar loss, in contrast to $\ell(\omega_k, \boldsymbol{\lambda})$, which is the individual loss or the vector loss, since it has a free index k.

B.3.4 Gradient of the loss

The simulation of the quantum sensor, the particle filter, and the evaluation of the NN are implemented in the chosen automatic differentiation (AD) environment, i.e. TensorFlow (TF), so that at the end of the simulation we can take the gradient of the loss in Eq. (B.60) with respect to $\boldsymbol{\lambda}$ with no effort and obtain

$$\frac{\mathrm{d}\mathcal{L}(\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} = \frac{1}{B}\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\sum_{k=1}^{B} \ell(\omega_k, \boldsymbol{\lambda}) \ . \tag{B.61}$$

The automatic differentiation framework does all the derivatives automatically, that we would need to evaluate analytically or numerically otherwise. Even if the outcomes $\boldsymbol{y}_{k,M-1}$ are extracted from a probability distribution that depends on $\boldsymbol{\lambda}$, as it is because each of them is sampled from $p(y_{k,t+1}|\boldsymbol{x}_{k,t+1}, \boldsymbol{y}_t, \boldsymbol{\theta}_k)$ and the controls $\boldsymbol{x}_{k,t+1}$ depend on $\boldsymbol{\lambda}$, in TF and other similar frameworks their derivatives with respect to $\boldsymbol{\lambda}$ are always null by construction, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} y_{k,t+1} = 0 \ , \tag{B.62}$$

in other words, the gradient cannot propagate through the extraction of random variables. This is a consistent behaviour of automatic differentiation

frameworks, and has to do with the fact that the sampled variables are considered constant tensors in the construction of the graph, on the same level as other numerical constants fixed by the programmer. We will show now, that much like in [PPM23], the gradient of the loss produced by AD in Eq. (B.61) is not correct and will lead to a suboptimal training routine. Another term must be added that keeps track of the sampled variables during the evolution. To understand why this is so, let us start from the theoretical definition of $\Delta^2\widehat{\boldsymbol{\theta}}$ in Eq. (B.58) and take its gradient with respect to $\boldsymbol{\lambda}$. The two terms $p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})$ and $\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1},\boldsymbol{\lambda})$ both depend on $\boldsymbol{\lambda}$. The first one can be expanded as follows:

$$p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda}) = \prod_{t=0}^{M-1} p(y_t|\boldsymbol{x}_t,\boldsymbol{y}_{t-1},\boldsymbol{\theta}) \ , \tag{B.63}$$

and the dependence on the controls $\boldsymbol{x}_t$ is a dependence on the trainable variables of the agent $\boldsymbol{\lambda}$. The second term $\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1},\boldsymbol{\lambda})$ depends on $\boldsymbol{\lambda}$ through the PF weights, which are updated with the Bayes rule Eq. (5.7), that features the term $p(y_{t+1}|\boldsymbol{x}_{t+1},\boldsymbol{y}_t,\boldsymbol{\theta})$, where again the controls $\boldsymbol{x}_{t+1}$ are $\boldsymbol{\lambda}$-dependent. The complete gradient of the right-hand term of Eq. (B.58) reads therefore

$$\int \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1},\boldsymbol{\lambda}),\boldsymbol{\theta})p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})\,\mathrm{d}\boldsymbol{y}_{M-1} + \int \ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1},\boldsymbol{\lambda}),\boldsymbol{\theta})\frac{\mathrm{d}p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}}\,\mathrm{d}\boldsymbol{y}_{M-1} \ . \tag{B.64}$$

The first term is in the form of an expectation value and can be straightforwardly approximated in a Monte Carlo simulation. It corresponds exactly to the naïve gradient of the loss in Eq. (B.61) computed by the AD framework. The second term can be written as

$$\int \ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1},\boldsymbol{\lambda}),\boldsymbol{\theta})\frac{\mathrm{d}\log p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}}p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})\,\mathrm{d}\boldsymbol{y}_{M-1} \ , \tag{B.65}$$

which is now in the form of an expectation value on the trajectories of the simulation and can be evaluated simultaneously with the first term, provided we keep track of $\log p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta},\boldsymbol{\lambda})$. This second contribution to the gradient can be approximated as

$$\frac{1}{B}\sum_{k=1}^{B}\ell(\omega_k,\boldsymbol{\lambda})\frac{\mathrm{d}\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} \ , \tag{B.66}$$

on a batch of B simulations. The term $\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda})$ is the sum

$$\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda}) = \sum_{t=0}^{M-1}\log p(y_{k,t}|\boldsymbol{y}_{k,t-1},\boldsymbol{\theta}_k,\boldsymbol{\lambda}) \ , \tag{B.67}$$

where we exchanged the dependence on $\boldsymbol{x}_{k,t}$ of the factors $p(y_{k,t}|\boldsymbol{x}_{k,t}, \boldsymbol{y}_{k,t-1}, \boldsymbol{\theta}_k)$ for the dependence on $\boldsymbol{\lambda}$. This logarithm can be accumulated step by step in the simulation, after the extraction of each measurement outcome. Notice that for B simulations in the batch, we have to compute B cumulated probabilities, because each trajectory is different. In conclusion, the total gradient is

$$\frac{1}{B}\sum_{k=1}^{B}\left[\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\ell(\omega_k,\boldsymbol{\lambda}) + \ell(\omega_k,\boldsymbol{\lambda})\frac{\mathrm{d}\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}}\right] . \tag{B.68}$$

By introducing the *stop gradient* operation we can write it in the convenient form

$$\frac{1}{B}\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\sum_{k=1}^{B}\{\ell(\omega_k,\boldsymbol{\lambda}) + \mathrm{sg}\left[\ell(\omega_k,\boldsymbol{\lambda}),\boldsymbol{\theta}\right]\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda})\} , \tag{B.69}$$

that requires only one gradient, which makes it more straightforward to implement in the AD framework. In this formula we are using the stop gradient operator $\mathrm{sg}\left[\cdot\right]$, which is an instruction that tells the automatic differentiation frameworks not to compute the derivatives of the expression inside the operator. We are therefore naturally led to introducing the modified loss $\widetilde{\mathcal{L}}(\boldsymbol{\lambda})$, i.e.

$$\widetilde{\mathcal{L}}(\boldsymbol{\lambda}) := \frac{1}{B}\sum_{k=1}^{B}\widetilde{\ell}(\omega_k,\boldsymbol{\lambda}) , \tag{B.70}$$

with

$$\widetilde{\ell}(\omega_k,\boldsymbol{\lambda}) := \ell(\omega_k,\boldsymbol{\lambda}) + \mathrm{sg}\left[\ell(\omega_k,\boldsymbol{\lambda})\right]\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k,\boldsymbol{\lambda}) . \tag{B.71}$$

which is the correct function to be minimized. For a resource limited estimation the modified loss is the average of the B' simulations that have terminated. In Section B.4.1 we comment on the gradient descent process, on the choice of the hyper-parameters, and the typical behaviour of the loss in the training. The second term of the gradient in Eq. (B.69) is similar to the loss of the Policy Gradient method in reinforcement learning [SMSM99] (RL), where the probabilities arise because of the stochastic extraction of the policy, while here the NN produces directly the action and the stochasticity comes from the measurement outcome extraction. The necessity of introducing such terms when dealing with the gradient of expressions involving non-reparametrizable random variables has been known in the Machine Learning literature for a while [FWF19] and the expressions involving stop-gradient operators go under

the name of surrogate expressions. However, the first time this has appeared in the physics literature is in [PPM23], applied to quantum feedback. Had we neglected the log-likelihood term of Eq. (B.69) we would have introduced a bias in the gradient. Not adding the log-likelihood terms means not only a slower convergence in the training but possibly also converging to a worse minimum.

Log-likelihood terms in the loss

We can generalize and say that whatever extraction of random variables we perform during the simulation, we need to add a corresponding log-likelihood term, but only if the probability distribution from which they have been extracted depends on $\boldsymbol{\lambda}$, implicitly or explicitly. With growing number of extracted variables the variance of the gradient grows and if the batchsize is too small this can severely affect the training, then we might loose convergence and end up in a bad local minimum. If possible we advice to reparametrize the random variables and account for the backpropagation of the gradient in a more direct way. Depending on the quantum sensor we are simulating we might be able to implement the extraction of the measurement outcomes through a differentiable reparametrization. If we can write the measurement outcome y_t as

$$y_t = g(u_t, x_t, \boldsymbol{\theta}) \ , \tag{B.72}$$

where u_t is a random variable extracted from a probability distribution independent on $\boldsymbol{\lambda}$, i.e. $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} p(u_t) = 0$, then we can omit the corresponding log-likelihood term in Eq. (B.69). The gradient propagates now directly through the measurement outcome, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} y_t \neq 0 \ , \tag{B.73}$$

and we can differentiate the loss in Eq. (B.60) as it is. The log-likelihood term would be $\log p(u_t)$, which is independent on $\boldsymbol{\lambda}$. The reparametrization can however be applied only to continuous variables, see [PPM23] for more details. In Appendix B.2 we apply this technique in the resampling step of the particle filter.

Adding a baseline

We can also try to add a baseline to Eq. (B.69), as suggested in [WT01] for RL with Policy Gradient. This means modifying the gradient to

$$\frac{1}{B} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \sum_{k=1}^{B} \left[\ell(\omega_k, \boldsymbol{\lambda}) + \mathrm{sg}\left[\ell(\omega_k, \boldsymbol{\lambda}) - \mathcal{B} \right] \log p(\boldsymbol{y}_{k,M-1} | \boldsymbol{\theta}_k, \boldsymbol{\lambda}) \right] \ , \tag{B.74}$$

with the standard choice for the baseline $\mathcal{B}$ being

$$\mathcal{B} := \frac{1}{B}\sum_{k=1}^{B} \ell(\omega_k, \boldsymbol{\lambda}) \ , \tag{B.75}$$

that is, inside the stop gradient we subtract to each loss in the batch the mean value of the loss on the batch. It is important for $\mathcal{B}$ to be a constant across the simulations indexed by k. We briefly see in the following that the introduction of $\mathcal{B}$ doesn't change the expected value gradient, while it can be proved that it reduces the variance of the gradient [WT01]. Consider the following chain of equalities

$$\begin{aligned} 0 &= \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \int p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \, \mathrm{d}\boldsymbol{y}_{M-1} \\ &= \int \frac{1}{p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda})} \frac{\mathrm{d}p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \, \mathrm{d}\boldsymbol{y}_{M-1} \ , \end{aligned} \tag{B.76}$$

where the first one comes from the normalization of $p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda})$ and in the second we divided and multiplied for $p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda})$ upon swapping the integral and the derivate. The rightmost term of Eq. (B.76) is now in the form of an expectation value, that can be approximated by the following summation in the simulation:

$$\begin{aligned} \int \frac{1}{p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda})} \frac{\mathrm{d}p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} p(\boldsymbol{y}_{M-1}|\boldsymbol{\theta}, \boldsymbol{\lambda}) \, \mathrm{d}\boldsymbol{y}_{M-1} \simeq \\ \frac{1}{B}\sum_{k=1}^{B} \frac{1}{p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda})} \frac{\mathrm{d}p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} \ . \end{aligned} \tag{B.77}$$

Where the term in the right-hand summation is the derivative of the log-likelihood, so that we expect

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \sum_{k=1}^{B} \log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda}) \simeq 0 \ , \tag{B.78}$$

for large B. Adding the baseline in Eq. (B.75), means adding a terms proportional to the derivative of the log-likelihood, with the proportionality constant being $\mathcal{B}$, which has null expectation value.

Discrete control space

In this section we briefly comment on the case in which the control space is discrete, that is, x_t can be chosen only among finitely many elements,

i.e $x_t \in \chi = \{x_1, x_2, \dots, x_R\}$. This happens for example in the experiment presented in [CPB+23], where the control parameter was the topological charge of the q-plate. In this case the agent produces a probability distribution on the set χ as outcome, just like in Policy Learning, and a random x_t is extracted from this categorical distribution. In this scenario we need to revisit the Eq. (B.56) and Eq. (B.57), that now need to accommodate also for the probability of extracting a particular x_t:

$$P(\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1}|\boldsymbol{\theta}) = \prod_{t=0}^{M-1} p(y_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1}, \boldsymbol{\theta}) g(x_t|\boldsymbol{y}_{t-1}, \boldsymbol{x}_{t-1}, \boldsymbol{\lambda}) \tag{B.79}$$

$$P(\widehat{\boldsymbol{\theta}}|\boldsymbol{x}_{M-1}, \boldsymbol{y}_{M-1}) = \delta(\widehat{\boldsymbol{\theta}} - g_2(\boldsymbol{y}_{M-1}, \boldsymbol{x}_{M-1})) \ . \tag{B.80}$$

Substituting this expressions in Eq. (B.51) we get

$$\Delta^2\widehat{\boldsymbol{\theta}} = \int \ell(\widehat{\boldsymbol{\theta}}(\boldsymbol{y}_{M-1}, \boldsymbol{x}_{M-1}), \boldsymbol{\theta}) \prod_{t=0}^{M-1} p(y_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1}, \boldsymbol{\theta}) g(x_t|\boldsymbol{y}_{t-1}, \boldsymbol{x}_{t-1}, \boldsymbol{\lambda}) \, \mathrm{d}\boldsymbol{x}_{M-1} \, \mathrm{d}\boldsymbol{y}_{M-1} \ . \tag{B.81}$$

By repeating the derivation of the loss, considering that $p(y_t|\boldsymbol{x}_t, \boldsymbol{y}_{t-1}, \boldsymbol{\theta})$ doesn't depend on $\boldsymbol{\lambda}$ anymore, the log-likelihood term of Eq. (B.69) becomes

$$\sum_{t=0}^{M-1} \mathrm{sg}\left[\ell(\omega_k, \boldsymbol{\lambda})\right] \log g(x_{k,t}|\boldsymbol{y}_{t-1,k}, \boldsymbol{x}_{t-1,k}, \boldsymbol{\lambda}) \ . \tag{B.82}$$

Stochastic estimator

In all the applications of this work the estimator $\widehat{\boldsymbol{\theta}}$ is always a deterministic function of the PF ensemble. Even for the case of values discrimination, this is computed as the most likely hypothesis at the end of the experiment, which doesn't require any sampling. A perfectly valid estimator for θ would be one sample extracted from the Bayesian posterior $P(\boldsymbol{\theta}|\boldsymbol{y}_{M-1}, \boldsymbol{\lambda})$. Doing so requires adding a term to the log-likelihood term in the loss in Eq. (B.69), which now becomes

$$\sum_{t=0}^{M-1} \mathrm{sg}\left[\ell(\omega_k, \boldsymbol{\lambda})\right] \left[\log p(\boldsymbol{y}_{k,M-1}|\boldsymbol{\theta}_k, \boldsymbol{\lambda}) + \log P(\widehat{\boldsymbol{\theta}}_k|\boldsymbol{y}_{k,M-1}, \boldsymbol{\lambda})\right] \ . \tag{B.83}$$

A differentiable expression for the posterior distribution at $\widehat{\boldsymbol{\theta}}$ might not be accessible if the parameters are continuous, but if they are discrete, the term $P(\widehat{\boldsymbol{\theta}}|\boldsymbol{y}_{M-1}, \boldsymbol{\lambda})$ is just the weight corresponding to the discrete values $\widehat{\boldsymbol{\theta}}$.

B.3.5 Definition of the cumulative and logarithmic losses

When doing a simulation for a certain M, if we want the result of the training to give us the optimal strategy also for $M_2 < M$ we can introduce the cumulative loss, that also takes into account the loss at intermediate steps. A naïve approach is to extend the MSE to all steps between $t = 0$ and $t = M - 1$, and write

$$\mathcal{L}_{\text{cum}}(\boldsymbol{\lambda}) := \frac{1}{MB} \sum_{t=0}^{M-1} \sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) \ , \tag{B.84}$$

where $\widehat{\boldsymbol{\theta}}_{k,t}$ is the estimator at step t of the k-th simulation. With this loss the agent is incentivized to make the estimator $\widehat{\boldsymbol{\theta}}$ converge to the value $\boldsymbol{\theta}$ as soon as possible. However the error on the first time steps of the estimation dominates the later errors in the summation, and this puts pressure on the agent to optimize the first steps of the procedure at the expense of the later precision. To solve this problem we divide each terms in the sum Eq. (B.84) by a function $\eta(\boldsymbol{\theta}, t)$, i.e.

$$\mathcal{L}_{\text{cum}}(\boldsymbol{\lambda}) := \frac{1}{MB} \sum_{t=0}^{M-1} \sum_{k=1}^{B} \frac{\ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k)}{\eta(\boldsymbol{\theta}_k, t)} \ , \tag{B.85}$$

where $\eta(\boldsymbol{\theta}_k, t)$ is the expected precision of the estimation at step t given the true value $\boldsymbol{\theta}_k$, or an approximation to it, in the form of a lower bound for example, like the Cramér-Rao bound. This new loss measures the relative variation of the error from the reference value. Even if $\eta(\boldsymbol{\theta}_k, t)$ is a rigorous lower bound on the MSE we can't expect the inequality

$$\ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) \geq \eta(\boldsymbol{\theta}_k, t) \ , \tag{B.86}$$

to hold exactly for every t and k, as there will be fluctuations due to the finite batchsize. From the practical point of view this means that it is possible for the loss of some training steps to be $\mathcal{L}(\boldsymbol{\lambda}) < 1$, which doesn't necessarily point toward a bug in the implementation of the training. With Eq. (B.85) we still incentivise the agent to be as fast as possible in reaching a good precision, and not wait until the end, because then it will be rewarded by the reduced loss for all the duration of the experiment. Another possibility to account fairly for the MSE at intermediate times is to take the logarithm of the mean error on the batch and write the cumulative loss as

$$\mathcal{L}_{\log}(\boldsymbol{\lambda}) := \frac{1}{M} \sum_{t=0}^{M-1} \log \left[\frac{1}{B} \sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) \right] \ . \tag{B.87}$$

The advantage of this approach is that it doesn't require any prior known reference value for the error. Notice that this loss is not in the form of an expectation value of $\ell(\widehat{\boldsymbol{\theta}}, \boldsymbol{\theta})$ over a batch.

B.3.6 Cumulative and logarithmic losses for a resource limited estimation

In this section we comment on the form taken by the cumulative and logarithmic losses in the case of a limited number of resources. Given M' the realized number of iterations of the measurement loop Eq. (B.85) becomes

$$\mathcal{L}_{\text{cum}}(\boldsymbol{\lambda}) := \frac{1}{M'B} \sum_{t=0}^{M'-1} \sum_{k=1}^{B} \frac{\ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k)}{\eta(\boldsymbol{\theta}_k, t)} \,, \tag{B.88}$$

notice, that at difference with Eq. (B.54) all the simulations in the batch are considerer, not only those B' that were already ended as the measurement loop stopped. If one estimation in the batch is ended prematurely with respect to all the other, with all the resources consumed to obtain a bad estimator for $\boldsymbol{\theta}$ this will have a huge weight in the loss, since the squared error will appear multiple times, until all the other estimations are ended. This means that an unwise use of the resources, which are consumed early to reach a poor result will be strongly penalized. One may think, that since the number of iterations M' is stochastic then teh cumulative loss is a form of "existential loss" which would put pressure on th agent to terminate with the smallest number of measurement step possible, this would be at odd with the actual goal of optimizing with fixed resources irrespective of the number of measurements, but indeed the loss is normalized according to M', so that having a short or a long cycle doesn't matter for the computation of $\mathcal{L}(\boldsymbol{\lambda})$. Similarly to the cumulative loss, the logarithmic loss for an estimation with a limited number of resources can be expressed as

$$\mathcal{L}_{\log}(\boldsymbol{\lambda}) := \frac{1}{M'} \sum_{t=0}^{M'-1} \log \left[\frac{1}{B} \sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) \right] \,, \tag{B.89}$$

where again M' is the actual number of executed iterations of the loop.

B.3.7 Gradients of the cumulative and logarithmic losses

In this section we comment on the expression of the gradient of the cumulative and logarithmic losses, and of the role of the log-likelihood terms

that we had inserted in Eq. (B.71). The modified cumulative loss, from which the AD framework can directly compute the gradient, reads

$$\widetilde{\mathcal{L}}_{\rm cum}(\boldsymbol{\lambda}) := \frac{1}{MB}\sum_{k=1}^{B}\Big\{\sum_{t=0}^{M-1}\ell(\widehat{\boldsymbol{\theta}}_{k,t},\boldsymbol{\theta}_k)+{\rm sg}\left[\sum_{t=0}^{M-1}\ell(\widehat{\boldsymbol{\theta}}_{k,t},\boldsymbol{\theta}_k)\right]\sum_{t=0}^{M-1}\log p(y_{k,t}|\boldsymbol{\theta}_k,\boldsymbol{y}_{t-1,k},\boldsymbol{\lambda})\Big\}\,. \tag{B.90}$$

Given that the stop gradient operator is linear, we now make the important observation that the gradient of the log-likelihood terms in the form

$${\rm sg}\left[\ell(\widehat{\boldsymbol{\theta}}_{\alpha,k},\boldsymbol{\theta}_k)\right]\log p(y_{\beta,k}|\boldsymbol{\theta}_k,\boldsymbol{y}_{\beta-1,k},\boldsymbol{\lambda})\,. \tag{B.91}$$

with $\beta>\alpha$ have null expectation value on the batch of simulations, that is

$$\frac{1}{B}\sum_{k=1}^{B}\ell(\widehat{\boldsymbol{\theta}}_{\alpha,k},\boldsymbol{\theta}_k)\frac{{\rm d}\log p(y_{\beta,k}|\boldsymbol{\theta}_k,\boldsymbol{y}_{\beta-1,k},\boldsymbol{\lambda})}{{\rm d}\boldsymbol{\lambda}}\simeq 0\,, \tag{B.92}$$

The expression in Eq. (B.92) is an approximation of the true expectation value

$$\int\ell(\widehat{\boldsymbol{\theta}}_{\alpha},\boldsymbol{\theta})\frac{{\rm d}\log p(y_{\beta}|\boldsymbol{\theta},\boldsymbol{y}_{\beta-1},\boldsymbol{\lambda})}{{\rm d}\boldsymbol{\lambda}}\pi(\boldsymbol{\theta})d\boldsymbol{\theta}\prod_{t=0}^{\beta}p(y_t|\boldsymbol{\theta},\boldsymbol{y}_{t-1},\boldsymbol{\lambda})\,{\rm d}y_t\,. \tag{B.93}$$

All the integral for y_t for $t>\beta$ can be simplified in the above formula, since the integrand doesn't depend on these variables. Let us first solve the integral for ${\rm d}y_\beta$. The loss term doesn't depend on this variable, so that we can pull it out of the integral and write

$$\ell(\widehat{\boldsymbol{\theta}}_{\alpha},\boldsymbol{\theta})\int\frac{{\rm d}\log p(y_{\beta}|\boldsymbol{\theta}_k,\boldsymbol{y}_{\beta-1},\boldsymbol{\lambda})}{{\rm d}\boldsymbol{\lambda}}p(y_\beta|\boldsymbol{\theta},\boldsymbol{y}_{t-1},\boldsymbol{\lambda})\,{\rm d}y_\beta\,, \tag{B.94}$$

which is equal to

$$\int\frac{{\rm d}p(y_{\beta}|\boldsymbol{\theta},\boldsymbol{y}_{t-1},\boldsymbol{\lambda})}{{\rm d}\boldsymbol{\lambda}}\,{\rm d}y_\beta=\frac{\rm d}{{\rm d}\boldsymbol{\lambda}}\int p(y_\beta|\boldsymbol{\theta},\boldsymbol{y}_{t-1},\boldsymbol{\lambda})\,{\rm d}y_\beta=0\,. \tag{B.95}$$

Since the summation Eq. (B.92) tends to zero for large B, we can write the loss as following

$$\widetilde{\mathcal{L}}_{\rm cum}(\boldsymbol{\lambda}) := \frac{1}{MB}\sum_{k=1}^{B}\Big\{\sum_{t=0}^{M-1}\ell(\widehat{\boldsymbol{\theta}}_{k,t},\boldsymbol{\theta}_k)+\sum_{t=0}^{M-1}{\rm sg}\left[\ell(\widehat{\boldsymbol{\theta}}_{k,t},\boldsymbol{\theta}_k)\right]\log p(\boldsymbol{y}_{k,t}|\boldsymbol{\theta}_k,\boldsymbol{\lambda})\Big\}\,. \tag{B.96}$$

which is the expression implemented in the library. Since we have removed some of the stochastic terms in the loss, which average to zero, but nevertheless contribute to the fluctuations, using expression Eq. (B.96) we expect to have reduced the variance of the gradient, just like we did with the correction of Ścibior and Wood for the particle resampling. From this derivation we learn that in general the log-likelihood terms of variables extracted in the future with respect to the terms they multiply can be simplified. Notice that in this derivation we haven't assumed projective measurements, that would have meant $p(y_t|\boldsymbol{\theta}_k, \boldsymbol{y}_{t-1}, \boldsymbol{\lambda}) = p(y_t|\boldsymbol{\theta}_k, \boldsymbol{\lambda})$, instead our derivation works in the most general case of a weakly measured probe. We now turn to the gradient of the logarithm loss of Eq. (B.89). This is somewhat different from the previous cases since now the mean on the batch is inside the logarithm. The expectation value of the loss is

$$\frac{1}{M}\sum_{t=0}^{M-1} \log\left[\int \ell(\widehat{\boldsymbol{\theta}}_t, \boldsymbol{\theta}) p(\boldsymbol{y}_t|\boldsymbol{\theta}, \boldsymbol{\lambda})\, \mathrm{d}\boldsymbol{y}_t\right] , \tag{B.97}$$

which has the following gradient

$$\frac{1}{M}\sum_{t=0}^{M-1} \frac{\int \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\ell(\widehat{\boldsymbol{\theta}}_t, \boldsymbol{\theta}) p(\boldsymbol{y}_t|\boldsymbol{\theta}, \boldsymbol{\lambda})\, \mathrm{d}\boldsymbol{y}_t + \int \ell(\widehat{\boldsymbol{\theta}}_t, \boldsymbol{\theta}) \frac{\mathrm{d}\log p(\boldsymbol{y}_t|\boldsymbol{\theta},\boldsymbol{\lambda})}{\mathrm{d}\boldsymbol{\lambda}} p(\boldsymbol{y}_t|\boldsymbol{\theta}, \boldsymbol{\lambda})\, \mathrm{d}\boldsymbol{y}_t}{\int \ell(\widehat{\boldsymbol{\theta}}_t, \boldsymbol{\theta}) p(\boldsymbol{y}_t|\boldsymbol{\theta}, \boldsymbol{\lambda})\, \mathrm{d}y_t} . \tag{B.98}$$

This expression can be obtained on the batch of simulations with the modified loss

$$\widetilde{\mathcal{L}}_{\log}(\boldsymbol{\lambda}) := \frac{1}{M}\sum_{t=0}^{M-1} \frac{\sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k) + \sum_{k=1}^{B} \mathrm{sg}\left[\ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k)\right] \log p(\boldsymbol{y}_{k,t}|\boldsymbol{\theta}_k, \boldsymbol{\lambda})}{\mathrm{sg}\left[\sum_{k=1}^{B} \ell(\widehat{\boldsymbol{\theta}}_{k,t}, \boldsymbol{\theta}_k)\right]} . \tag{B.99}$$

To get the results for the resources limited estimation we substitute M with M' in the whole section.

B.4 Details on the simulations

B.4.1 Choice and tuning of the hyperparameters

In this section, we briefly comment on the choice of hyperparameters for the examples reported here. These include the batchsize B, the number of particles in the PF N, and the initial learning rate α_0. They must be chosen according to the memory limitations of the computer during training. Specifically, we first empirically fix the number of particles to a value large

enough to ensure that the discretization of the posterior does not compromise the precision of the estimation. This, in turn, determines the batchsize, i.e., the number of simulations that can be executed in parallel. The batchsize, along with the type of loss, then determines the initial learning rate. For example, in Fig. 5.2, we used $B = 128$ and $\alpha_0 = 10^{-2}$ for the cumulative loss, as well as $B = 1024$ and $\alpha_0 = 10^{-3}$ for the logarithmic loss. These values are used for the time and measurements-limited estimations respectively. The batchsize can also be "artificially" increased via gradient accumulation, which involves averaging the gradients from multiple executions of a batch of simulations for the update in Eq. (5.20). For the Dolinar receiver, we used $\alpha = 10^{-2}$ and $B = 4096$.

We mentioned earlier that the update of the agent's trainable variables is not actually done through Eq. (5.20), but via a more sophisticated optimizator called Adam. We observed empirically that a decaying learning rate is beneficial when using the Adam optimizer. This is because the agent first learns the rough features of the optimal solution with a relatively large update step for the variables. Subsequently, with a smaller learning rate, the solution is fine tuned. The Adam optimized, however, already has an internal adaptive update step that is different for every variables, therefore the learning rate should be really only understood as a broad indication of the training speed given to the optimizer. In the original Adam paper [KB15] the authors consider a learning rate decaying with the inverse square root of the number of update steps. This was also our choice. Let us define $i = 1, 2, \cdots, I$ the index of the update step in the training process, then the learning rate at the i-th iteration of the gradient descent is

$$\alpha_i := \frac{\alpha_0}{\sqrt{i}} \,. \tag{B.100}$$

We observe empirically, that the initial value of the learning rate α_0 for a NN should depend on the batchsize B. For $B \sim \mathcal{O}(10^3)$ we use $\alpha_0 \sim \mathcal{O}(10^{-2})$, while for $B \sim \mathcal{O}(10^2)$ a value of $\alpha_0 \sim \mathcal{O}(10^{-3})$ is more appropriate. For the non-adaptive strategy we use an initial learning rate that is one order of magnitude larger than the one used for the NN at equal batchsize. The minimum number of training steps I depend strongly on the application, but we observed in all our examples that it should of order $I \sim \mathcal{O}(10^3 - 10^4)$ to reach convergence. We observed some universal feature in the behaviour of the loss as the training proceeds, which can be associated to three different phases in the training, see Fig. B.2. First we have an initial phase of fast learning, which is the shortest one, coloured in pink, followed by the fine tuning phase in yellow and the plateau at the end, with the loss remaining on average constant. As a final note, we mention that when the resampling

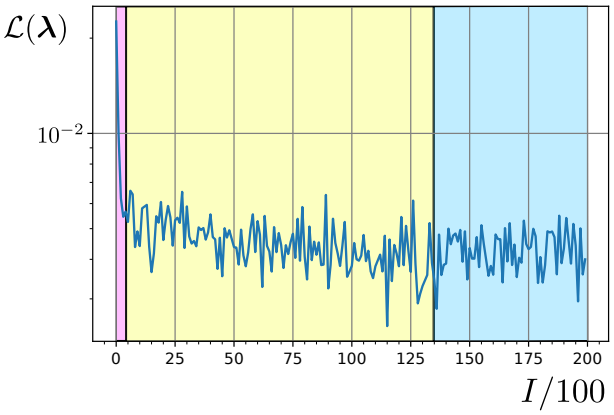
routine is active we might expect a slow-down of the simulation speed as the training session proceeds. This happens because as the agent is perfected and the loss is reduced, it is more probable that a resampling event is triggered (because the increasing precision means also more concentrated weights in the PF ensemble), which slows down the simulation. In other words, the amount of code that has to be executed in a run is not fixed *a priori*, but depends dynamically on the resampling condition that is checked at run-time. To end the section we briefly recap the three possible implementations that the trainable variables $\boldsymbol{\lambda}$ have taken in this work, see Fig. B.2. In the case the agent is a NN $\boldsymbol{\lambda}$ are the weights and the biases of the network When the agent is a decision tree, the controls x_t are associated to each node of the tree, and they are the $\boldsymbol{\lambda}$ variables. Finally, for a non-adaptive strategy, there is no adaptivity and the controls $\boldsymbol{x}$ are codified in a list, indexed by the measurement step t. In this case the controls and the training variables coincide, i.e. $\boldsymbol{\lambda} = \boldsymbol{x}_{M-1}$.

B.4.2 Fit of the precision

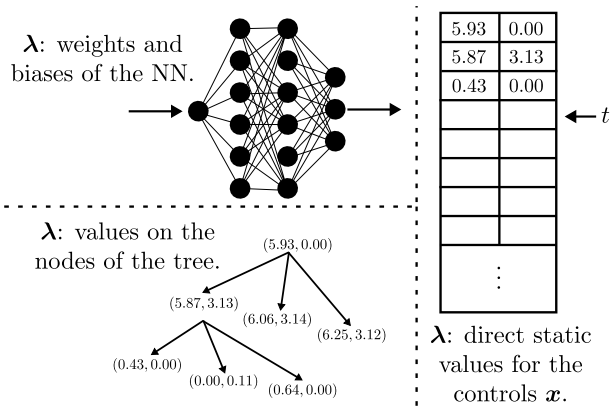
In this section we briefly comment on the way the precision plot are realized throughout the work. The definition of the resources doesn't only impact the stopping condition of the measurement loop, but it defines how the performances of an agent are visualized, since by default we plot the mean loss as a function of the consumed resources. After having trained the agent we simulate many times the estimation and we keep track of the tuples $\mathcal{S} := \{(R_j, \mathcal{L}_j)\}_{j=1}^{S}$ after each measurement step, containing the consumed resources R_t and the loss $\mathcal{L}$. Since the experiment is a stochastic process we will collect a cloud of points from which a simple relation between the expected precision and the resources must be obtained, see Fig. B.3. The first possibility is to divide the x-axis of the resources in intervals of size δ, and compute the barycenters of all the points $(R_j, \mathcal{L}_j)$ falling in this interval, these would be the red crosses of Fig. B.3. The second possibility is to fit this cloud of points with a NN. We set the training loss to be the MSE, i.e.

$$\mathcal{L}_{\text{fit}} := \frac{1}{S} \sum_{j=1}^{S} \left[\mathcal{L}_j - f_{\text{NN}}(R_j)\right]^2 \ , \tag{B.101}$$

which, for a single value of the resource, i.e. $R_j = R \ \forall j$, would converge to $f_{\text{NN}}(R) = \frac{1}{S}\sum_{j=1}^{S} \mathcal{L}_j$, that is a NN will approximate the mean loss. This won't be exactly true for a cloud of points, but with Eq. (B.101) we incentivise the NN to converge toward the average loss for every value of the resources. All the plots in this paper have been produced with the first method, choosing

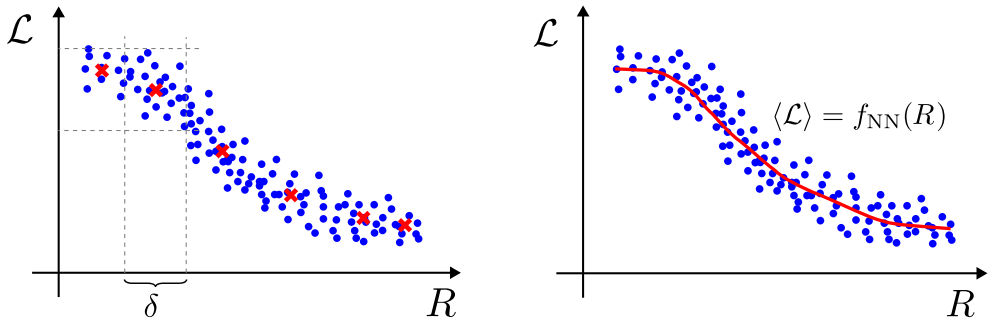


(a) Three phases of the training.



(b) Implementations of the trainable variables.

Figure B.2: On the left picture the three phases of the learning process explained in Section B.4.1 are plotted. One tick on the x-axis corresponds to 500 update steps. The agent first learns the rough form of the optimal strategy and later the fine details, before converging. On the right the three agents used in this work are represented, i.e a NN, a ternary decision tree, and a table containing the values of the controls $\boldsymbol{x}$, indexed by the measurement step t.



(a) Clustering of the precision point cloud.

(b) Fit of the precision point cloud.

Figure B.3: On the left side we represent the precision plot where the cloud of points has been clustered to obtain the red crosses. On the right we use a NN to interpolate and get the average loss at a given value of the resources.

an appropriate δ, except for those plots on the NV center platform with $T_2^\star = \infty$ and referring to the time-limited estimation. In the PGH line of the first plot in Fig. 5.2 there is a non-monotonicity for small T, that is a defect in the plot and an artifact on this way of interpolating with a NN.

B.4.3 Scaling of the time and memory requirements

Since the B estimations in a batch can be performed in parallel we will benefit from the use of a GPU or a TPU (Tensor Processing Unit) in the training of the agent. The main difference between the CPU and the GPU is that a CPU has fewer ($\sim \mathcal{O}(10)$) faster cores, while a GPU has many ($\sim \mathcal{O}(10^3)$) slower cores. With a large batchsize the use of hardware acceleration through a GPU will turn out to be essential and we will first examine the resource requirements of model-aware RL assuming that everything that can be parallelized has been is indeed executed in parallel. In this case the time requirement of the simulation is mainly influenced linearly by the number of measurements M in the training loop, that have to be executed necessarily sequentially. The update of the PF and the computation of the distributions moments all require $\mathcal{O}(N)$ multiplications each but can be done in parallel, where N is the number of particles. The memory requirement depends on the batchsize B and the number of particles N in the PF. Nevertheless because of the construction of the gradient, for which we need to keep in memory the results of all the intermediate computations, the number of measurements M has also an almost linear influence on the required memory. Finally, the total time used in the training is also proportional to the number of update

steps I. Each update step comprises the complete run of an estimation batch together with the evaluation of the gradient and the update of the controls. The size of the NN has little impact on the training time and memory. We can summarise the above considerations as

$$\text{Memory} \sim \mathcal{O}(BMN) \,, \quad \text{Time}_{\text{Par}} \sim \mathcal{O}(IM) \,. \tag{B.102}$$

Assuming that nothing can be parallelized (we have a single core) and therefore everything is sequential, if, as usual, the computational time in the CPU is dominated by the number of floating point multiplications, we instead have the time scaling

$$\text{Time}_{\text{Seq}} \sim \mathcal{O}(IBMN) \,, \tag{B.103}$$

while the memory requirement is unchanged. Neither a GPU nor a CPU will perfectly reproduce these theoretical scalings, because the GPU has a limited number of cores, but there is a tendency for a GPU to follow the scaling of Time_{Par} and for a CPU Time_{Seq}. If the batchsize B is very large (or the GPU not very powerful) the simulations in the batch can't all be executed in parallel and B starts to affect also the time requirements. If B and N are small a CPU may complete the training before a GPU, because of the smaller proportionality factor for the time requirement in Eq. (B.103) with respect to Eq. (B.102), due to the faster cores of the CPU. This analysis applies also to the training of a non-adaptive strategy, which is not resource-saving compared to the training of the NN. In the applications we expect our agent to run on a small controller near the sensor, where most definitely we won't have access to a GPU and lots of memory, which anyway are required only in the training phase. In this situation we have no batch and only one iteration, i.e. $I = B = 1$. Furthermore there is no extra M in the memory requirement appearing because of the automatic differentiation, so that the resource scaling in the application will be

$$\text{Memory} \sim \mathcal{O}(dN + N) \,, \quad \text{Time}_{\text{Seq}} \sim \mathcal{O}(MN) \,, \tag{B.104}$$

where d is the number of parameters. For an estimation limited by the resources instead of the number of measurements, M must be intended as the average number of measurements employed for a fixed amount of resources. In general thanks to the resampling routine we can keep the number of particles fixed while increasing the precision arbitrarily. It is a good practise although to choose N proportional to the number of parameters to estimate, i.e. $N \sim \mathcal{O}(d)$. In the applications the memory requirement of the NN, and the multiplications needed to evaluate it at each step contribute respectively with additional $\mathcal{O}(n_l n_u)$ memory and $\mathcal{O}(n_l n_u^2)$ time (per step), where n_l is

the number of layers and n_u the number of units per layer, so that we have

$$\text{Memory} \sim \mathcal{O}(dN + N + n_l n_u) \ , \quad \text{Time}_{\text{Seq}} \sim \mathcal{O}(MN + M n_l n_u^2) \ . \tag{B.105}$$

If the control is non-adaptive we don't need this extra computations, and if the PF is removed from the picture (because we don't need real time feedback) we have that the memory and time requirements trivialize, i.e they scale respectively as $\mathcal{O}(1)$ and $\mathcal{O}(M)$. Of course the total time of estimation would be influenced also by the time it takes to perform the physical measurement on the probe, but here we are referring only to the computational time.

B.5 Backward recursion method for the optimization of the strategy

In this section we set the stage to understand what function the agent must approximate by formulating the problem in terms of a backward recursion. In this section we will see how the optimal control could theoretically be derived in other ways, so that the training will appear less as an unintelligible black-box and more as the solution to a well-posed problem (though we won't probably have uniqueness). The output of the agent at the $t+1$-th steps is x_{t+1}, that is, the control of the current evolutions and measurements. In the following we will define formally the function that the agent must learn to approximate, in doing this we will assume that the control x of the quantum sensor is a continuos real variable. Consider an estimation with M measurements, i.e. $t = 0, 1, \ldots, M-1$. Let us focus on the last one only and recall the definition of the particle filter ensemble before after the last measure $M-1$, i.e. $\mathfrak{p}_M := \{\boldsymbol{\theta}_j^{M-1}, w_j^{M-1}\}_{j=1}^N$. Then we can write the ensemble of the PF at the final step $t = M-1$ as

$$\mathfrak{p}_M = \mathfrak{B}(\mathfrak{p}_{M-1}, x_{M-1}, y_{M-1}) \ , \tag{B.106}$$

where $\mathfrak{B}$ encodes the application of the Bayes rule in Eq. (5.7). The ensemble $\mathfrak{p}_M$ inherits the stochasticity from the random measurement outcome y_{M-1}. Per definition $\mathfrak{p}_0$ is the initial PF ensemble that represents the prior $\pi(\boldsymbol{\theta})$. In Section B.3.1 we mentioned that the final loss is a scalar function $\ell(\mathfrak{p}_M, \boldsymbol{\theta})$ of the final PF ensemble and of the true value $\boldsymbol{\theta}$, like the squared derivation of the estimator from the true value. The final loss can be expressed as $\ell(\mathfrak{B}(\mathfrak{p}_{M-1}, x_{M-1}, y_{M-1}), \boldsymbol{\theta})$ and it's expectation value on y_{M-1} (the expected loss), computed with the density in Eq. (5.1), reads

$$\bar{\ell}(\mathfrak{p}_{M-1}, x_{M-1}, \boldsymbol{\theta}) := \int \ell(\mathfrak{B}(\mathfrak{p}_{M-1}, x_{M-1}, y_{M-1}), \boldsymbol{\theta}) p(y_{M-1}|x_{M-1}, \boldsymbol{\theta}) dy_{M-1} \ . \tag{B.107}$$

If the outcome probability, the prior and the loss are continuos functions we can also expect $\overline{\ell}(\mathfrak{p}_{M-1}, x_{M-1}, \boldsymbol{\theta})$ to be continuous in its parameters. Without aiming at full rigour, we say that the regularity properties of the probability densities are passed down to the expectation value. Now we look for the minimum of this function, which is realized by solving

$$\frac{\mathrm{d}\overline{\ell}(\mathfrak{p}_{M-1}, x^{\star}_{M-1}, \boldsymbol{\theta})}{\mathrm{d}x_{M-1}} = 0 \, . \tag{B.108}$$

This equation defines implicitly the optimal control $x^{\star}_{M-1} := r_{M-1}(\mathfrak{p}_{M-1}, \boldsymbol{\theta})$, where $x^{\star}_{M-1}$ realizes the absolute minimum of the expected loss. r_{M-1} inherits some regularity property (at least locally) from $\overline{\ell}(\mathfrak{p}_{M-1}, x_{M-1}, \boldsymbol{\theta})$ thanks to the implicit function theorem. The control $x^{\star}_{M-1}$ can still have discontinuities in $\mathfrak{p}_{M-1}$ if the expected loss has multiple competing minima. The dependence on $\boldsymbol{\theta}$ is rather inconvenient, because it is unknown, but we can think of substituting $\boldsymbol{\theta}$ with its estimator $\widehat{\boldsymbol{\theta}}_{M-2}$ to get $x^{\star}_{M-1} = r_{M-1}(\mathfrak{p}_{M-1}, \widehat{\boldsymbol{\theta}}_{M-2})$. We will however never do explicit optimizations with this approach, the introduction of Machine Learning in quantum metrology serves precisely to avoid these cumbersome computations. Until now we have only optimized the last control, but fortunately all these operations can be repeated with minor changes for the $t = M - 2$ measurement step. Let us start from $\mathfrak{p}_{M-1}$ expressed as function of the ensemble $\mathfrak{p}_{M-2}$:

$$\mathfrak{p}_{M-1} = \mathfrak{B}(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}) \, , \tag{B.109}$$

we insert this expression in Eq. (B.106) to get

$$\mathfrak{p}_{M} = \mathfrak{B}(\mathfrak{B}(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}), x_{M-1}, y_{M-1}) \, . \tag{B.110}$$

By substituting this in the loss $\ell(\mathfrak{p}_M, \boldsymbol{\theta})$ we get $\ell(\mathfrak{B}(\mathfrak{B}(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}), x_{M-1}, y_{M-1}), \boldsymbol{\theta})$, but the optimal x_{M-1} has been already computed as a function of the PF ensemble. Whatever control we suggest for the step $t = M - 2$ it doesn't change the optimal control at the following step. In other words whatever control x_{M-2} gets applied, the optimal action for the last time step is always $x^{\star}_{M-1}$, we can therefore insert it in the loss at the step $M - 2$, i.e.

$$\begin{aligned} &\ell(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}, y_{M-1}, \boldsymbol{\theta}) := \\ &\ell(\mathfrak{B}(\mathfrak{B}(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}), r_{M-1}(\mathfrak{B}(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}), \boldsymbol{\theta}), y_{M-1}), \boldsymbol{\theta}) \, , \end{aligned} \tag{B.111}$$

where we have also used the expression for $\mathfrak{p}_{M-1}$ of Eq. (B.109) in the definition of $x^{\star}_{M-1}$, and we have redefined the parameters of ℓ. Again we take

the expectation value on the measurement outcomes, i.e.

$$\overline{\ell}(\mathfrak{p}_{M-2}, x_{M-2}, \boldsymbol{\theta}) := \int \ell(\mathfrak{p}_{M-2}, x_{M-2}, y_{M-2}, y_{M-1}, \boldsymbol{\theta}) \\ p(y_{M-1}|\boldsymbol{\theta}, x^\star_{M-1}) p(y_{M-2}|\boldsymbol{\theta}, x_{M-2}) dy_{M-1} dy_{M-2} \,. \quad \text{(B.112)}$$

By taking the derivative of this expected loss with respect to x_{M-2}, we define implicitly the optimal control $x^\star_{M-2} = r_{M-2}(\mathfrak{p}_{M-2}, \boldsymbol{\theta})$ as the solution of

$$\frac{\mathrm{d}\overline{\ell}(\mathfrak{p}_{M-2}, x^\star_{M-2}, \boldsymbol{\theta})}{\mathrm{d}x_{M-2}} = 0 \,, \quad \text{(B.113)}$$

where $x^\star_{M-2}$ realizes the absolute minimum of $\overline{\ell}$. Notice that this is not a greedy optimization: the value of x_{M-2} is not chosen to optimize the loss one step head in the future but the final loss, knowing what the strategy in the next step will be. We could treat all the previous measurements in the same manner, if it wasn't for the expectation values of the loss, that become more and more complicated. In this way we can inductively proceed in reverse to the start of the estimation $t = 0$ and find in the process the family of functions $r_t(\mathfrak{p}_t, \boldsymbol{\theta}_{t-1})$ that express the optimal controls $x^\star_t$. As discussed previously we can redefine $r_t(\mathfrak{p}_t) = r_t(\mathfrak{p}_t, \widehat{\boldsymbol{\theta}}_{t-1})$, in order to get rid of the dependence on the unknown value of the parameters $\boldsymbol{\theta}$. We then introduce $r(\mathfrak{p}_t, t) = r_t(\mathfrak{p}_t)$, which is the function that the agent is trying to approximate in the training, i.e. the map that spits out the optimal control at a given measurement step t, provided the ensemble PF at the previous step. Heuristically we expect the optimal control to be inhomogeneous in time because like in many application of RL a good strategy encompasses a phase of "exploration" followed by a phase of "exploitation" of what has been learned [Mar21a, Mar21b].

B.6 Backpropagation of the gradient

With the help of an automatic differentiation framework we compute the gradient of the modified loss in Eq. (B.70) in order to perform the training. Let us for the moment neglect the log-likelihood terms in this expression and concentrate only on the first part. If the loss is computed only from the ensemble at the last measurement step, then it takes the form

$$\ell_{\boldsymbol{\theta}} \circ \mathfrak{B}_{x_{M-1}, y_{M-1}} \circ \mathfrak{B}_{x_{M-1}, y_{M-1}} \circ \cdots \circ \mathfrak{B}_{x_0, y_0}(\mathfrak{p}_0) \,. \quad \text{(B.114)}$$

where $\ell_{\boldsymbol{\theta}}(\mathfrak{p}) := \ell(\mathfrak{p}, \boldsymbol{\theta})$ acts on the ensemble PF and is the individual loss of each simulation, e.g. the square error. The function $\mathfrak{B}_{x_t, y_t}(\mathfrak{p}) := \mathfrak{B}(\mathfrak{p}, x_t, y_t)$

applies the Bayesian update to the PF, which depends on the outcome of the measurement y_t and on the control x_t. The initial ensemble of the PF is named $\mathfrak{p}_0$. From a theoretical point of view differentiating this expression means applying repeatedly the rule for the derivation of the composite function and propagating the derivative through the various stages of the estimation, this is called backpropagation. Let us compute explicitly the derivative of the loss for the last two steps of the estimation. We define $\mathfrak{p}_M = \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})$ and $\mathfrak{p}_{M-1} = \mathfrak{B}_{x_{M-2},y_{M-2}}(\mathfrak{p}_{M-2})$, and apply the chain rule. We will indicate with the partial derivative symbol the derivatives with respect to the parameters of a function (which can also appear in the subscript), while the symbol $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}$ is reserved to the total derivative with respect to $\boldsymbol{\lambda}$.

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}}\ell_{\boldsymbol{\theta}}(\mathfrak{p}_M) = \frac{\partial \ell_{\boldsymbol{\theta}}(\mathfrak{p}_M)}{\partial \mathfrak{p}} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1}) \,. \tag{B.115}$$

The total derivate in the right-hand term an be expanded again with the chain rule:

$$\frac{\partial \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})}{\partial x_{M-1}} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1}) + \underline{\frac{\partial \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})}{\partial \mathfrak{p}} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-2},y_{M-2}}(\mathfrak{p}_{M-2})} \quad \text{a)} \,. \tag{B.116}$$

Only the parameters x_{M-1} and $\mathfrak{p}_{M-1}$ can carry a dependence on $\boldsymbol{\lambda}$. Regarding the measurement outcomes y_{M-1} we already discussed their independence on $\boldsymbol{\lambda}$, expressed by Eq. (B.62). The control x_{M-1} has an explicit dependence on $\boldsymbol{\lambda}$ because it has been produced by the agents, but also has a dependence on $\boldsymbol{\lambda}$ through the PF ensemble, so that we can expand the total derivative in the following way

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1}) = \underline{\frac{\partial x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1})}{\partial \boldsymbol{\lambda}}} \quad \text{b)} \tag{B.117}$$

$$+ \underline{\frac{\partial x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1})}{\partial \mathfrak{p}} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-2},y_{M-2}}(\mathfrak{p}_{M-2})} \quad \text{c)} \,. \tag{B.118}$$

The first piece of the derivative is the dependence on $\boldsymbol{\lambda}$ that comes from the last application of the agent, while the second piece represent the backpropagation through the input of the agent and the Bayesian update of the PF ensemble. In general, the gradient is backpropagated through all the applications of the agent until it reaches the beginning. We notice that we can write the total derivative of $\mathfrak{p}_M$ as a function of the total derivative of $\mathfrak{p}_{M-1}$, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1}) = Q_{M-1} + H_{M-1} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-2},y_{M-2}}(\mathfrak{p}_{M-2}) \tag{B.119}$$

where

$$Q_{M-1} := \frac{\partial \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})}{\partial x_{M-1}} \frac{\partial x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1})}{\partial \boldsymbol{\lambda}} , \tag{B.120}$$

$$H_{M-1} := \frac{\partial \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})}{\partial x_{M-1}} \frac{\partial x_{M-1}(\boldsymbol{\lambda}, \mathfrak{p}_{M-1})}{\partial \mathfrak{p}} + \frac{\partial \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1})}{\partial \mathfrak{p}} . \tag{B.121}$$

We have arrived to a family of recurrence equations, which have the solution

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\lambda}} \mathfrak{B}_{x_{M-1},y_{M-1}}(\mathfrak{p}_{M-1}) = \sum_{t=0}^{M-1} Q_t \prod_{m=t+1}^{M-1} H_m , \tag{B.122}$$

where Q_t and H_m are defined analogously to Q_{M-1} and H_{M-1}. The H_m terms have each two summand, and when multiplied together the number of terms in the gradient grows exponentially in the number of measurement M, and generates gradient terms corresponding to multiple repeated backpropagations through the agents. This are a kind of "higher order" gradient terms. In our implementation of the training we simplify the gradient by introducing a stop gradient operation before the input of the agent, so that Eq. (5.10) is actually implemented as

$$x_{t+1} = \mathcal{F}\{\mathrm{sg}\,[P(\boldsymbol{\theta}|\boldsymbol{x}_t, \boldsymbol{y}_t); R_t; t]\} , \tag{B.123}$$

This modification doesn't change the forward pass, that is, the results of simulations are the same, but it affects the backpropagation of the gradient, in particular it makes the first term of H_m disappear, because

$$\frac{\partial x_m\left(\boldsymbol{\lambda}, \mathrm{sg}\,[\mathfrak{p}_m]\right)}{\partial \mathfrak{p}} = 0 . \tag{B.124}$$

Such simplification reduces considerably the training time and doesn't really affect, at least from a theoretical point of view, the ability of the agent to learn the optimal strategy, on the contrary it might even improve it. Before we back up this last assertions we want to recapitulate our analysis of the backpropagation with the help of Fig. B.4. A control strategy is called myopic if it optimizes the information gained from the next measurement only, while it is non-myopic if it optimizes many steps ahead in the future. It might seem that by cutting the gradient propagation through the green arrows we limit the optimization to the class of myopic strategies, but this is not true because the optimization is always done for the final precision. The input of the agent is basically a constant now and the t-th term in the summation

of Eq. (B.122) pulls the weights of the NN to minimize the final loss given the PF ensemble at the $t-1$-th step. In order to have non-myopic adaptive strategy backpropagating the gradient through the green arrows is redundant. The gradient of the log-likelihood terms in the loss of Eq. (B.71) is also simplified when the stop gradient is acting in Eq. (B.124). In the model $p(y_t|x_t, \boldsymbol{\theta})$ the only term that depends on $\boldsymbol{\lambda}$ is the control, and the gradient propagates through a single application of the agent. Had we not inserted the stop gradient, the gradient of each summands in the log-likelihood would have been propagated through all the past applications of the agent, a scenario that happens anyway if the probe state is not reinitialized between measurements. Before ending the discussion on the gradient backpropagation we want to consider yet another possibility of truncating the gradient. We could indeed image to stop the flow of the derivative through the evolution of the PF ensemble, which means cutting the red line in Eq. (B.116). This eliminates the recurrence equation and trivializes the gradient, which now accounts only for the very last control. That is, with such modification, the agent will learn to optimize only the last measurement. If the loss is cumulative however, like in Eq. (B.84), all the controls will be optimized, but in a myopic way. In this scenario we introduce the modified logarithmic loss, i.e.

$$\widetilde{\mathcal{L}}_{\log}(\boldsymbol{\lambda}) := \frac{1}{TB}\sum_{t=1}^{M-1}\log\left[\frac{\sum_{k=1}^{B}\ell(\widehat{\boldsymbol{\theta}}_{k,t},\boldsymbol{\theta}_k)}{\mathrm{sg}\left[\sum_{k=1}^{B}\ell(\widehat{\boldsymbol{\theta}}_{t-1,k},\boldsymbol{\theta}_k)\right]}\right] . \tag{B.125}$$

Each term in this summation is the empirical information gain for a Gaussian posterior. In the forward pass the loss is the total empirical information gain, because the stop gradient operators don't play any role, and the series can be resummed. In the computation of the gradient, however, the information gain for each measurement is optimized greedily, as done in [GFWC12] and in the package optbayesexpt [MBD21]. If the stop gradient in the denominator is applied only every n measurement, then we are optimizing the information gain planning n steps ahead in the future. It is advisable to put a regularization at the denominator of the loss in Eq. (B.125) to avoid dividing by zero.

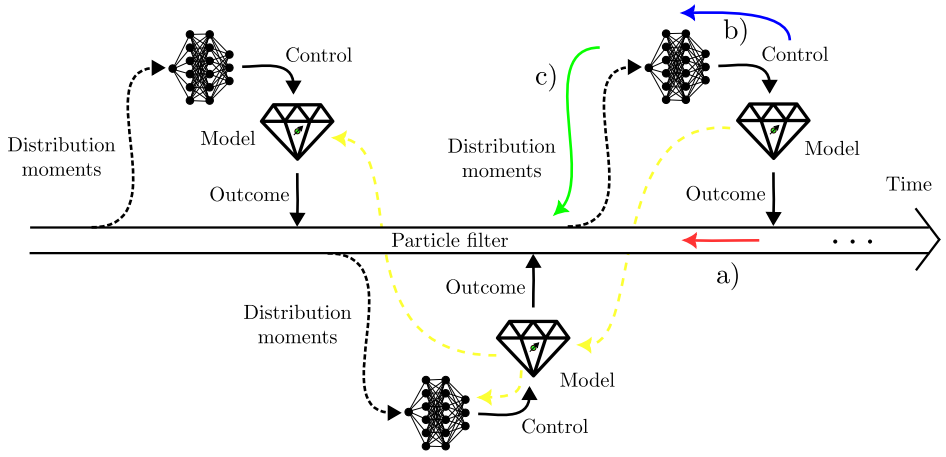


Figure B.4: The fat empty arrow in the middle of the picture represents the PF, which is updated via the Bayes rule, indicated with the function $\mathfrak{B}$ in the text, after it receive an outcome from the measurement on the probe. The second term of Eq. (B.116), underlined in red and labelled with a), corresponds to the backpropagation of the gradient along the history of the PF ensemble, whose weights and particles inherits a dependence on $\boldsymbol{\lambda}$ from the actions of the agent. Visually this corresponds to backpropagation along the red arrow labelled with a). Through the match between the underlined terms in Eq. (B.118) and Eq. (B.116) and the arrows in the figure can visualize the origin of each term of the gradient. The blue term of Eq. (B.118) labelled with b) accounts for the dependence on $\boldsymbol{\lambda}$ that comes from the controls computed by the agent and propagated through the application of the Bayes rule and the computation of x_t. The green term of Eq. (B.118) labelled with c) is the gradient propagating from the input of the agent to the previous PF ensemble. This term is responsible for the "higher-order" terms of the gradient, that propagate multiple times through the agent. Inserting the stop gradient in Eq. (B.124) means cutting the green line. The dashed yellow line represents the propagation of the gradient through the state of the probe, when this is not reinitialized between the measurements.

Chapter 7

Bibliography

[ABH+13] Luigi Ambrosio, Alberto Bressan, Dirk Helbing, Axel Klar, Enrique Zuazua, Luigi Ambrosio, and Nicola Gigli. A user's guide to optimal transport. *Modelling and Optimisation of Flows on Networks: Cetraro, Italy 2009, Editors: Benedetto Piccoli, Michel Rascle*, pages 1–155, 2013.

[ABH+22] Muhammad Junaid Arshad, Christiaan Bekker, Ben Haylock, Krzysztof Skrzypczak, Daniel White, Benjamin Griffiths, Joe Gore, Gavin W. Morley, Patrick Salter, Jason Smith, Inbar Zohar, Amit Finkler, Yoann Altmann, Erik M. Gauger, and Cristian Bonato. Online adaptive estimation of decoherence timescales for a single qubit, October 2022. arXiv:2210.06103 [quant-ph].

[ACC+21] Andrew Arrasmith, M. Cerezo, Piotr Czarnik, Lukasz Cincio, and Patrick J. Coles. Effect of barren plateaus on gradient-free optimization. *Quantum 5, 558 (2021)*, (5), Oct 2021.

[ADPP11a] Antonio Assalini, Nicola Dalla Pozza, and Gianfranco Pierobon. Revisiting the dolinar receiver through multiple-copy state discrimination theory. *Phys. Rev. A*, 84:022342, Aug 2011.

[ADPP11b] Antonio Assalini, Nicola Dalla Pozza, and Gianfranco Pierobon. Revisiting the Dolinar receiver through multiple-copy state discrimination theory. *Physical Review A*, 84(2):022342, August 2011.

[Agr13] Julián Agredo. A Wasserstein-type distance to measure deviation from equilibrium of quantum Markov semigroups. *Open Systems & Information Dynamics*, 20(02):1350009, 2013.

[Agr16] Julián Agredo. On exponential convergence of generic quantum Markov semigroups in a Wasserstein-type distance. *International Journal of Pure and Applied Mathematics*, 107(4):909–925, 2016.

[AH11] Daiki Akimoto and Masahito Hayashi. Discrimination of the change point in a quantum setting. *Physical Review A - Atomic, Molecular, and Optical Physics*, 83(5):052328, may 2011.

[AK22] Eric R Anschuetz and Bobak T Kiani. Quantum variational algorithms are swamped with traps. *Nature Communications*, 13(1):7760, 2022.

[Amb08] Luigi Ambrosio. Gradient flows in metric spaces and in the spaces of probability measures, and applications to fokker-planck equations with respect to log-concave measures. *Bollettino dell'Unione Matematica Italiana*, 1(1):223–240, 2 2008.

[AMGC02] M.S. 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*, 50(2):174–188, February 2002.

[Ban94] Masashi Ban. Quasicontinuous measurements of photon number. *Phys. Rev. A*, 49:5078–5085, Jun 1994.

[Ban99] Konrad Banaszek. Optimal receiver for quantum cryptography with two coherent states. *Physics Letters A*, 253(1):12–15, 1999.

[BB17] Cristian Bonato and Dominic W. Berry. Adaptive tracking of a time-varying field with a quantum sensor. *Physical Review A*, 95(5):052348, May 2017.

[BBD+16] C. Bonato, M. S. Blok, H. T. Dinani, D. W. Berry, M. L. Markham, D. J. Twitchen, and R. Hanson. Optimized

quantum sensing with a single electron spin using real-time adaptive measurements. *Nature Nanotechnology*, 11(3):247–252, March 2016.

[BBMO21] Zachary Ballard, Calvin Brown, Asad M. Madni, and Aydogan Ozcan. Machine learning and computation-enabled intelligent sensor design. *Nature Machine Intelligence*, 3(7):556–565, July 2021.

[BC09] Stephen M. Barnett and Sarah Croke. Quantum state discrimination. *Adv. Opt. Photon.*, 1(2):238–278, Apr 2009.

[BCM+21] Atılım Güneş Baydin, Kyle Cranmer, Pablo de Castro Manzano, Christophe Delaere, Denis Derkach, Julien Donini, Tommaso Dorigo, Andrea Giammanco, Jan Kieseler, Lukas Layer, Gilles Louppe, Fedor Ratnikov, Giles C. Strong, Mia Tosi, Andrey Ustyuzhanin, Pietro Vischia, and Hevjin Yarar. Toward Machine Learning Optimization of Experimental Design. *Nuclear Physics News*, 31(1):25–28, January 2021.

[BDS+18] Marin Bukov, Alexandre G. R. Day, Dries Sels, Phillip Weinberg, Anatoli Polkovnikov, and Pankaj Mehta. Reinforcement Learning in Different Phases of Quantum Control. *Physical Review X*, 8(3):031086, September 2018.

[BED+21] Yue Ban, Javier Echanobe, Yongcheng Ding, Ricardo Puebla, and Jorge Casanova. Neural-network-based parameter estimation for quantum detection. *Quantum Science and Technology*, 6(4):045012, October 2021.

[Ber07] János A Bergou. Quantum state discrimination and selected applications. *Journal of Physics: Conference Series*, 84:012001, oct 2007.

[Ber10] János A. Bergou. Discrimination of quantum states. *Journal of Modern Optics*, 57(3):160–180, 2010.

[BEŻ22] Rafał Bistroń, Michał Eckstein, and Karol Życzkowski. Monotonicity of the quantum 2-Wasserstein distance. *arXiv:2204.07405*, 2022.

[BFH06] János A. Bergou, Edgar Feldman, and Mark Hillery. Optimal unambiguous discrimination of two subspaces as a case in mixed-state discrimination. *Physical Review A - Atomic, Molecular, and Optical Physics*, 73(3):032107, mar 2006.

[BH05] János A. Bergou and Mark Hillery. Universal programmable quantum state discriminator that is optimal for unambiguously distinguishing between unknown states. *Physical Review Letters*, 94(16):160501, apr 2005.

[BIS+18] Sergio Boixo, Sergei V. Isakov, Vadim N. Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J. Bremner, John M. Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, apr 2018.

[BIS+22] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, Shahnawaz Ahmed, Vishnu Ajith, M. Sohaib Alam, Guillermo Alonso-Linaje, B. AkashNarayanan, Ali Asadi, Juan Miguel Arrazola, Utkarsh Azad, Sam Banning, Carsten Blank, Thomas R. Bromley, Benjamin A. Cordier, Jack Ceroni, Alain Delgado, Olivia Di Matteo, Amintor Dusko, Tanya Garg, Diego Guala, Anthony Hayes, Ryan Hill, Aroosa Ijaz, Theodor Isacsson, David Ittah, Soran Jahangiri, Prateek Jain, Edward Jiang, Ankit Khandelwal, Korbinian Kottmann, Robert A. Lang, Christina Lee, Thomas Loke, Angus Lowe, Keri McKiernan, Johannes Jakob Meyer, J. A. Montañez-Barrera, Romain Moyard, Zeyue Niu, Lee James O'Riordan, Steven Oud, Ashish Panigrahi, Chae-Yeun Park, Daniel Polatajko, Nicolás Quesada, Chase Roberts, Nahum Sá, Isidor Schoch, Borun Shi, Shuli Shu, Sukin Sim, Arshpreet Singh, Ingrid Strandberg, Jay Soni, Antal Száva, Slimane Thabet, Rodrigo A. Vargas-Hernández, Trevor Vincent, Nicola Vitucci, Maurice Weber, David Wierichs, Roeland Wiersema, Moritz Willmann, Vincent Wong, Shaoming Zhang, and Nathan Killoran. PennyLane: Automatic differentiation of hybrid quantum-classical computations, July 2022. arXiv:1811.04968 [physics, physics:quant-ph].

[BLBSM23] Jessica Bavaresco, Patryk Lipka-Bartosik, Pavel Sekatski, and Mohammad Mehboudi. Designing optimal protocols in Bayesian quantum parameter estimation with higher-order operations, November 2023.

[BmrcvicvSDcv08] Lucie Bartůšková, Antonín Černoch, Jan Soubusta, and Miloslav Dušek. Programmable discriminator of coherent states: Experimental realization. *Phys. Rev. A*, 77:034306, Mar 2008.

[BRYC20a] M. Bilkis, M. Rosati, R. Morral Yepes, and J. Calsamiglia. Real-time calibration of coherent-state receivers: Learning by trial and error. *Phys. Rev. Research*, 2:033295, Aug 2020.

[BRYC20b] M. Bilkis, M. Rosati, R. Morral Yepes, and J. Calsamiglia. Real-time calibration of coherent-state receivers: Learning by trial and error. *Physical Review Research*, 2(3):033295, August 2020.

[BSB+20] John F Barry, Jennifer M Schloss, Erik Bauch, Matthew J Turner, Connor A Hart, Linh M Pham, and Ronald L Walsworth. Sensitivity optimization for NV-diamond magnetometry. *Rev. Mod. Phys.*, 92(1):68, 2020.

[BVM+21] Michael Broughton, Guillaume Verdon, Trevor McCourt, Antonio J. Martinez, Jae Hyeon Yoo, Sergei V. Isakov, Philip Massey, Ramin Halavati, Murphy Yuezhen Niu, Alexander Zlokapa, Evan Peters, Owen Lockwood, Andrea Skolik, Sofiene Jerbi, Vedran Dunjko, Martin Leib, Michael Streif, David Von Dollen, Hongxiang Chen, Shuxiang Cao, Roeland Wiersema, Hsin-Yuan Huang, Jarrod R. McClean, Ryan Babbush, Sergio Boixo, Dave Bacon, Alan K. Ho, Hartmut Neven, and Masoud Mohseni. TensorFlow Quantum: A Software Framework for Quantum Machine Learning, August 2021. arXiv:2003.02989 [cond-mat, physics:quant-ph].

[BŻ17] Ingemar Bengtsson and Karol Życzkowski. *Geometry of Quantum States: An Introduction to Quantum Entanglement*. Cambridge University Press, 2017.

[BZG24a] Federico Belliardo, Fabio Zoratti, and Vittorio Giovannetti. Application of machine learning to experimental design in quantum mechanics. *International Journal of Quantum Information*, January 2024.

[BZG24b] Federico Belliardo, Fabio Zoratti, and Vittorio Giovannetti. Applications of model-aware reinforcement learning in bayesian quantum metrology. *Phys. Rev. A*, 109:062609, Jun 2024.

[BZMG] Federico Belliardo, Fabio Zoratti, Florian Marquardt, and Vittorio Giovannetti. Model-aware reinforcement learning for high-performance bayesian experimental design in quantum metrology.

[CAB+21] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, Aug 2021.

[CDQ+17] C. Chen, D. Dong, B. Qi, I. R. Petersen, and H. Rabitz. Quantum ensemble classification: A sampling-based learning control approach. *IEEE Transactions on Neural Networks and Learning Systems*, 28(6):1345–1359, 2017.

[CEFŻ21] Sam Cole, Michał Eckstein, Shmuel Friedland, and Karol Życzkowski. Quantum Optimal Transport. *arXiv:2105.06922*, 2021.

[CGAB21] K Craigie, E M Gauger, Y Altmann, and C Bonato. Resource-efficient adaptive Bayesian tracking of magnetic fields with a quantum sensor. *Journal of Physics: Condensed Matter*, 33(19):195801, May 2021.

[CGNT17] Yongxin Chen, Tryphon T Georgiou, Lipeng Ning, and Allen Tannenbaum. Matricial Wasserstein-1 distance. *IEEE control systems letters*, 1(1):14–19, 2017.

[CGP20] Emanuele Caglioti, François Golse, and Thierry Paul. Quantum optimal transport is cheaper. *Journal of Statistical Physics*, 181(1):149–162, 2020.

[CGP21] Emanuele Caglioti, François Golse, and Thierry Paul. Towards Optimal Transport for Quantum Densities. *arXiv:2101.03256*, 2021.

[CGS+19] Valeria Cimini, Ilaria Gianani, Nicolò Spagnolo, Fabio Leccese, Fabio Sciarrino, and Marco Barbieri. Calibration of Quantum Sensors by Neural Networks. *Physical Review Letters*, 123(23):230502, December 2019.

[CGT18a] Yongxin Chen, Tryphon T Georgiou, and Allen Tannenbaum. Matrix optimal mass transport: a quantum mechanical approach. *IEEE Transactions on Automatic Control*, 63(8):2612–2619, 2018.

[CGT18b] Yongxin Chen, Tryphon T Georgiou, and Allen Tannenbaum. Wasserstein geometry of quantum states and optimal transport of matrix-valued measures. In *Emerging Applications of Control and Systems Theory*, pages 139–150. Springer, 2018.

[Che00] Anthony Chefles. Quantum state discrimination. *Contemporary Physics*, 41(6):401–424, 2000.

[CHH+22] Chaohan Cui, William Horrocks, Shuhong Hao, Saikat Guha, Nasser Peyghambarian, Quntao Zhuang, and Zheshen Zhang. Quantum receiver enhanced by adaptive learning. *Light: Science & Applications*, 11(1):344, December 2022.

[CLSS22] Chi-Ning Chou, Peter J. Love, Juspreet Singh Sandhu, and Jonathan Shi. Limitations of Local Quantum Algorithms on Random MAX-k-XOR and Beyond. In Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff, editors, *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*, volume 229 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 41:1–41:20, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[CM14] Eric A Carlen and Jan Maas. An analog of the 2-Wasserstein metric in non-commutative probability under which the Fermionic Fokker–Planck equation is gradient

flow for the entropy. *Communications in Mathematical Physics*, 331(3):887–926, 2014.

[CM17] Eric A Carlen and Jan Maas. Gradient flow and entropy inequalities for quantum Markov semigroups with detailed balance. *Journal of Functional Analysis*, 273(5):1810–1869, 2017.

[CM20] Eric A Carlen and Jan Maas. Non-commutative calculus, optimal transport and functional inequalities in dissipative quantum systems. *Journal of Statistical Physics*, 178(2):319–378, 2020.

[CMG07a] Robert L. Cook, Paul J. Martin, and J. M. Geremia. Optical coherent state discrimination using a closed-loop quantum measurement. *Nature*, 4 2007.

[CMG07b] Robert L. Cook, Paul J. Martin, and J. M. Geremia. Optical coherent state discrimination using a closed-loop quantum measurement. *Nature*, 446(7137):774–777, April 2007.

[CMZ+18] Ming Chen, Chao Meng, Qi Zhang, Changkui Duan, Fazhan Shi, and Jiangfeng Du. Quantum metrology with single spins in diamond under ambient conditions. *National Science Review*, 5(3):346–355, May 2018.

[Col12] A. J.T. Colina. Programmed discrimination of multiple sets of qbits with added classical information. *European Physical Journal D*, 66(7):185, jul 2012.

[Coy22] Brian Coyle. *Machine learning applications for noisy intermediate-scale quantum computers*. PhD thesis, University of Edinburgh, 2022.

[CPB+23] Valeria Cimini, Emanuele Polino, Federico Belliardo, Francesco Hoch, Bruno Piccirillo, Nicolò Spagnolo, Vittorio Giovannetti, and Fabio Sciarrino. Experimental metrology beyond the standard quantum limit for a wide resources range. *npj Quantum Information*, 9(1):1–9, March 2023.

[CPP+20] Fabio Cavaliere, Enrico Prati, Luca Poti, Imran Muhammad, and Tommaso Catuogno. Secure quantum communication technologies and systems: From labs to markets. *Quantum Reports*, 2(1):80–106, 2020.

[CSV+21] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 03 2021.

[CV20] Marco Chiani and Lorenzo Valentini. Short Codes for Quantum Channels With One Prevalent Pauli Error Type. *IEEE Journal on Selected Areas in Information Theory*, 1(2):480–486, 2020.

[CYL+19] Shouvanik Chakrabarti, Huang Yiming, Tongyang Li, Soheil Feizi, and Xiaodi Wu. Quantum wasserstein generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 6781–6792, 2019.

[DAM20] Sergey Dushenko, Kapildeb Ambal, and Robert D. McMichael. Sequential Bayesian Experiment Design for Optically Detected Magnetic Resonance of Nitrogen-Vacancy Centers. *Physical Review Applied*, 14(5):054036, November 2020.

[DB02] Miloslav Dušek and Vladimir Bužek. Quantum-controlled measurement device for quantum-state discrimination. *Physical Review A - Atomic, Molecular, and Optical Physics*, 66(2):1–5, aug 2002.

[DBG+19] Hossein T. Dinani, Dominic W. Berry, Raul Gonzalez, Jeronimo R. Maze, and Cristian Bonato. Bayesian estimation for quantum sensing in the absence of single-shot detection. *Physical Review B*, 99(12):125413, March 2019.

[DCEL09] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. Exact and approximate unitary 2-designs and their application to fidelity estimation. *Physical Review A*, 80(1):012304, jul 2009.

[DDF22] Marcus W. Doherty, Chunhui Rita Du, and Gregory D. Fuchs. Quantum science and technology based on color

centers with accessible spin. *Journal of Applied Physics*, 131(1):010401, January 2022.

[DM97] Pierre Del Moral. Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 325(6):653–658, September 1997.

[Dol73a] Jr. Dolinar, S. J. Processing and transmission of information. *Massachusetts Institute of Technology. Research Laboratory of Electronics. Quarterly Progress Report, no. 111*, 1973.

[Dol73b] Jr. Dolinar, S. J. Processing and transmission of information. *Massachusetts Institute of Technology. Research Laboratory of Electronics. Quarterly Progress Report, no. 111*, 1973.

[DPL14] Nicola Dalla Pozza and Nicola Laurenti. Adaptive discrimination scheme for quantum pulse-position-modulation signals. *Phys. Rev. A*, 89:012339, Jan 2014.

[DPMRF23] Giacomo De Palma, Milad Marvian, Cambyse Rouzé, and Daniel Stilck França. Limitations of variational quantum algorithms: a quantum optimal transport approach. *PRX Quantum*, 4(1):010309, 2023.

[DPMTL21] Giacomo De Palma, Milad Marvian, Dario Trevisan, and Seth Lloyd. The quantum wasserstein distance of order 1. *IEEE Transactions on Information Theory*, 67(10):6627–6643, 2021.

[DPP15] Nicola Dalla Pozza and Gianfranco Pierobon. Optimality of square-root measurements in quantum state discrimination. *Phys. Rev. A*, 91:042334, Apr 2015.

[DPR22] Giacomo De Palma and Cambyse Rouzé. Quantum Concentration Inequalities. *Annales Henri Poincaré*, 23:3391–3429, 2022.

[DPT21] Giacomo De Palma and Dario Trevisan. Quantum optimal transport with quantum channels. *Annales Henri Poincaré*, 22(10):3199–3234, 2021.

[DPT22] Giacomo De Palma and Dario Trevisan. The wasserstein distance of order 1 for quantum spin systems on infinite lattices. *arXiv:2210.11446*, 2022.

[DR20] Nilanjana Datta and Cambyse Rouzé. Relating relative entropy, optimal transport and Fisher information: A quantum HWI inequality. *Annales Henri Poincaré*, 21:2115–2150, 2020.

[DRC17] C. L. Degen, F. Reinhard, and P. Cappellaro. Quantum sensing. *Rev. Mod. Phys.*, 89:035002, Jul 2017.

[DRLM21] Tim De Ryck, Samuel Lanthaler, and Siddhartha Mishra. On the approximation of functions by tanh neural networks. *Neural Networks*, 143:732–750, November 2021.

[dSGD14] Marcus P. da Silva, Saikat Guha, and Zachary Dutton. Optimal discrimination of m coherent states with a small quantum computer. 2014.

[Duv21] Rocco Duvenhage. Optimal quantum channels. *Phys. Rev. A*, 104:032604, Sep 2021.

[EF01] Y. C. Eldar and G. D. Forney. On quantum detection and the square-root measurement. *IEEE Transactions on Information Theory*, 47(3):858–872, March 2001.

[EMV03] Y. C. Eldar, A. Megretski, and G. C. Verghese. Designing optimal quantum detectors via semidefinite programming. *IEEE Transactions on Information Theory*, 49(4):1007–1012, April 2003.

[EMV04] Y. C. Eldar, A. Megretski, and G. C. Verghese. Optimal detection of symmetric mixed quantum states. *IEEE Transactions on Information Theory*, 50(6):1198–1207, June 2004.

[FD04] Jaromír Fiurášek and Miloslav Dušek. Probabilistic quantum multimeters. *Physical Review A - Atomic, Molecular, and Optical Physics*, 69(3):032302, mar 2004.

[FDF02] Jaromír Fiurášek, Miloslav Dušek, and Radim Filip. Universal Measurement Apparatus Controlled by Quantum Software. *Physical Review Letters*, 89(19):190401, oct 2002.

[FECidZ22] Shmuel Friedland, Michał Eckstein, Sam Cole, and Karol Życzkowski. Quantum Monge-Kantorovich Problem and Transport Distance between Density Matrices. *Phys. Rev. Lett.*, 129:110402, Sep 2022.

[FGC12] Christopher Ferrie, Christopher E. Granade, and D. G. Cory. Adaptive Hamiltonian estimation using Bayesian experimental design. *AIP Conference Proceedings*, 1443(1):165–173, May 2012.

[FGC13] Christopher Ferrie, Christopher E. Granade, and D. G. Cory. How to best sample a periodic probability distribution, or on the accuracy of Hamiltonian finding strategies. *Quantum Information Processing*, 12(1):611–623, January 2013.

[FHJ+20] Fulvio Flamini, Arne Hamann, Sofiène Jerbi, Lea M Trenkwalder, Hendrik Poulsen Nautrup, and Hans J Briegel. Photonic architecture for reinforcement learning. *New Journal of Physics*, 22(4):045002, April 2020.

[FIMR21] Adam Foster, Desi R. Ivanova, Ilyas Malik, and Tom Rainforth. Deep Adaptive Design: Amortizing Sequential Bayesian Experimental Design. In *Proceedings of the 38th International Conference on Machine Learning*, pages 3384–3395. PMLR, July 2021.

[Fis35] R. A. Fisher. *The design of experiments*. The design of experiments. Oliver & Boyd, Oxford, England, 1935.

[FMG19] M. Fanizza, A. Mari, and V. Giovannetti. Optimal universal learning machines for quantum state discrimination. *IEEE Transactions on Information Theory*, 65(9):5931–5944, 2019.

[Fos21] A. E. Foster. *Variational, Monte Carlo and policy-based approaches to Bayesian experimental design*. http://purl.org/dc/dcmitype/Text, University of Oxford, 2021.

[FRTG22] Alessio Fallani, Matteo A. C. Rossi, Dario Tamascelli, and Marco G. Genoni. Learning Feedback Control Strategies for Quantum Metrology. *PRX Quantum*, 3(2):020310, April 2022.

[FSB21] Lukas J. Fiderer, Jonas Schuff, and Daniel Braun. Neural-Network Heuristics for Adaptive Bayesian Quantum Estimation. *PRX Quantum*, 2(2):020303, April 2021.

[FSW08] Andrew S. Fletcher, Peter W. Shor, and Moe Z. Win. Channel-adapted quantum error correction for the amplitude damping channel. *IEEE Transactions on Information Theory*, 54(12):5705–5718, 2008.

[FTWM18] Thomas Fösel, Petru Tighineanu, Talitha Weiss, and Florian Marquardt. Reinforcement Learning with Neural Networks for Quantum Feedback. *Physical Review X*, 8(3):031084, September 2018.

[FWF19] Gregory Farquhar, Shimon Whiteson, and Jakob Foerster. Loaded DiCE: Trading off Bias and Variance in Any-Order Score Function Gradient Estimators for Reinforcement Learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[Gal19] Adam Gali. Ab initio theory of the nitrogen-vacancy center in diamond. *Nanophotonics*, 8(11):1907–1943, November 2019.

[Gam08] Sébastien Gambs. Quantum classification, 2008.

[Ger04a] JM Geremia. Distinguishing between optical coherent states with imperfect detection. *Physical Review A*, 70(6):062303, December 2004.

[Ger04b] John M. Geremia. Distinguishing between optical coherent states with imperfect detection. *Phys. Rev. A*, 70:062303, Dec 2004.

[GFH+17] Christopher Granade, Christopher Ferrie, Ian Hincks, Steven Casagrande, Thomas Alexander, Jonathan Gross, Michal Kononenko, and Yuval Sanders. QInfer: Statistical inference software for quantum applications. *Quantum*, 1:5, April 2017.

[GFK+21] Antonio A. Gentile, Brian Flynn, Sebastian Knauer, Nathan Wiebe, Stefano Paesani, Christopher E. Granade, John G. Rarity, Raffaele Santagati, and Anthony Laing.

Learning models of quantum systems from experiments. *Nature Physics*, 17(7):837–843, July 2021.

[GFWC12] Christopher E. Granade, Christopher Ferrie, Nathan Wiebe, and D. G. Cory. Robust online Hamiltonian learning. *New Journal of Physics*, 14(10):103013, October 2012.

[GIN18] Laszlo Gyongyosi, Sandor Imre, and Hung Viet Nguyen. A survey on quantum channel capacities. *IEEE Communications Surveys & Tutorials*, 20(2):1149–1205, 2018.

[Gio04] V. Giovannetti. Quantum-enhanced measurements: Beating the standard quantum limit. *Science*, 306(5700):1330–1336, November 2004.

[GJL20] Li Gao, Marius Junge, and Nicholas LaRacuente. Fisher information and logarithmic sobolev inequality for matrix-valued functions. *Annales Henri Poincaré*, 21(11):3409–3478, 2020.

[GK10] Mădălin Guţă and Wojciech Kotłowski. Quantum learning: asymptotically optimal classification of qubit states. *New Journal of Physics*, 12(12):123032, dec 2010.

[GLM06] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum metrology. *Phys. Rev. Lett.*, 96:010401, Jan 2006.

[GLM11] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Advances in quantum metrology. *Nature Photonics*, 5(4):222–229, Apr 2011.

[GMP16] François Golse, Clément Mouhot, and Thierry Paul. On the mean field and classical limits of quantum mechanics. *Communications in Mathematical Physics*, 343(1):165–205, 2016.

[Gol18] François Golse. The quantum N-body problem in the mean-field and semiclassical regime. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2118):20170229, 2018.

[Got97] Daniel Gottesman. Stabilizer codes and quantum error correction, 1997.

[Got09] Daniel Gottesman. An introduction to quantum error correction and fault-tolerant quantum computation, 2009.

[GP17] François Golse and Thierry Paul. The Schrödinger equation in the mean-field and semiclassical regime. *Archive for Rational Mechanics and Analysis*, 223(1):57–94, 2017.

[GP18] François Golse and Thierry Paul. Wave packets and the quadratic Monge–Kantorovich distance in quantum mechanics. *Comptes Rendus Mathematique*, 356(2):177–197, 2018.

[GRR19] T. Gefen, A. Rotem, and A. Retzker. Overcoming resolution limits with quantum sensing. *Nature Communications*, 10(1), November 2019.

[GRTZ02] Nicolas Gisin, Grégoire Ribordy, Wolfgang Tittel, and Hugo Zbinden. Quantum cryptography. *Reviews of Modern Physics*, 74(1):145–195, March 2002.

[GSG+23] Valentin Gebhart, Raffaele Santagati, Antonio Andrea Gentile, Erik M. Gauger, David Craig, Natalia Ares, Leonardo Banchi, Florian Marquardt, Luca Pezzè, and Cristian Bonato. Learning quantum systems. *Nature Reviews Physics*, 5(3):141–156, March 2023.

[GT07] Nicolas Gisin and Rob Thew. Quantum communication. *Nature Photonics*, 1(3):165–171, March 2007.

[Guh11] Saikat Guha. Structured optical receivers to attain superadditive capacity and the holevo limit. *Phys. Rev. Lett.*, 106:240502, Jun 2011.

[Ham50] R. W. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160, 1950.

[Hay05] Masahito Hayashi. *Asymptotic Theory of Quantum Statistical Inference: Selected Papers*. February 2005. Pages: 542.

[HB07] Bing He and János A. Bergou. Programmable unknown quantum-state discriminators with multiple copies of program and data: A Jordan-basis approach. *Physical Review A - Atomic, Molecular, and Optical Physics*, 75(3):032316, mar 2007.

[HB08] Ulrike Herzog and János A. Bergou. Optimum unambiguous identification of d unknown pure qudit states. *Physical Review A - Atomic, Molecular, and Optical Physics*, 78(3):032320, sep 2008.

[Hel69] Carl W. Helstrom. Quantum detection and estimation theory. *Journal of Statistical Physics*, 1(2):231–252, June 1969.

[Hel76a] Carl W. Helstrom. *Quantum detection and estimation theory / Carl W. Helstrom.* Academic Press New York, 1976.

[Hel76b] C.W. Helstrom. *Quantum Detection and Estimation Theory.* Mathematics in Science and Engineering : a series of monographs and textbooks. Academic Press, New York, 1976.

[HH03] B. Hall and B.C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction.* Graduate Texts in Mathematics. Springer, 2003.

[HHH05] A. Hayashi, M. Horibe, and T. Hashimoto. Quantum pure-state identification. *Phys. Rev. A*, 72:052306, Nov 2005.

[HHH06] A. Hayashi, M. Horibe, and T. Hashimoto. Unambiguous pure-state identification without classical knowledge. *Phys. Rev. A*, 73:012328, Jan 2006.

[HNC+22] Hsien-Yi Hsieh, Jingyu Ning, Yi-Ru Chen, Hsun-Chung Wu, Hua Li Chen, Chien-Ming Wu, and Ray-Kuang Lee. Direct Parameter Estimations from Machine Learning-Enhanced Quantum State Tomography. *Symmetry*, 14(5):874, May 2022.

[Hol73a] A. S. Holevo. Statistical problems in quantum physics. In G. Maruyama and Yu. V. Prokhorov, editors, *Proceedings of the Second Japan-USSR Symposium on Probability Theory*, pages 104–119, Berlin, Heidelberg, 1973. Springer Berlin Heidelberg.

[Hol73b] A. S. Holevo. Statistical problems in quantum physics. In G. Maruyama and Yu. V. Prokhorov, editors, *Proceedings of the Second Japan-USSR Symposium on Probability Theory*, Lecture Notes in Mathematics, pages 104–119, Berlin, Heidelberg, 1973. Springer.

[Hol11] Alexander S. Holevo. *Probabilistic and Statistical Aspects of Quantum Theory*. Monographs (Scuola Normale Superiore). 2011.

[HOR21] Daniel Herr, Benjamin Obert, and Matthias Rosenkranz. Anomaly detection with variational quantum generative adversarial networks. *Quantum Science and Technology*, 6(4):045004, jul 2021.

[HTS+23] A. Muñoz de las Heras, C. Tabares, J. T. Schneider, L. Tagliacozzo, D. Porras, and A. González-Tudela. Photonic quantum metrology with variational quantum optical non-linearities, September 2023.

[HWO+19] Stuart Hadfield, Zhihui Wang, Bryan O'Gorman, Eleanor Rieffel, Davide Venturelli, and Rupak Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34, Feb 2019.

[IFK+21] Desi R Ivanova, Adam Foster, Steven Kleinegesse, Michael U. Gutmann, and Thomas Rainforth. Implicit Deep Adaptive Design: Policy-Based Experimental Design without Likelihoods. In *Advances in Neural Information Processing Systems*, volume 34, pages 25785–25798. Curran Associates, Inc., 2021.

[Ike20] Kazuki Ikeda. Foundation of quantum optimal transport and applications. *Quantum Information Processing*, 19(1):25, 2020.

[ITF+12] Shuro Izumi, Masahiro Takeoka, Mikio Fujiwara, Nicola Dalla Pozza, Antonio Assalini, Kazuhiro Ema, and Masahide Sasaki. Displacement receiver for phase-shift-keyed coherent states. *Physical Review A*, 86(4):042328, October 2012.

[JRO+17] Peter D. Johnson, Jonathan Romero, Jonathan Olson, Yudong Cao, and Alán Aspuru-Guzik. Qvector: an algorithm for device-tailored quantum error correction, 2017.

[JSS+21] Timo Joas, Simon Schmitt, Raffaele Santagati, Antonio Andrea Gentile, Cristian Bonato, Anthony Laing, Liam P. McGuinness, and Fedor Jelezko. Online adaptive quantum characterization of a nuclear spin. *npj Quantum Information*, 7(1):1–8, April 2021.

[Kan06] L. V. Kantorovich. On the translocation of masses. *Journal of Mathematical Sciences*, 133:1381–1382, 03 2006.

[KB15] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[KB23] Emre Köse and Daniel Braun. Superresolution imaging with multiparameter quantum metrology in passive remote sensing. *Physical Review A*, 107(3):032607, March 2023.

[Ken73] R. S. Kennedy. A near-optimum receiver for the binary coherent state quantum channel. *MIT Research Laboratory of Electronics Quarterly Progress Report 108*, pages 219–225, 1973.

[KHL18] Peter Karkus, David Hsu, and Wee Sun Lee. Particle Filter Networks with Application to Visual Localization. In *Proceedings of The 2nd Conference on Robot Learning*, pages 169–178, October 2018. ISSN: 2640-3498.

[KLFM23] Mario Krenn, Jonas Landgraf, Thomas Foesel, and Florian Marquardt. Artificial intelligence and machine learning for quantum technologies. *Physical Review A*, 107(1):010101, January 2023.

[KLM22] Leeseok Kim, Seth Lloyd, and Milad Marvian. Hamiltonian Quantum Generative Adversarial Networks. *arXiv:2211.02584*, 2022.

[KLMN01a] E. Knill, R. Laflamme, R. Martinez, and C. Negrevergne. Benchmarking quantum computers: The five-qubit error correcting code. *Physical Review Letters*, 86(25):5811–5814, jun 2001.

[KLMN01b] E Knill, R Laflamme, R Martinez, and C Negrevergne. Implementation of the five qubit error correction benchmark. *arXiv preprint quant-ph/0101034*, 2001.

[KOSH99] K. Kato, M. Osaki, M. Sasaki, and O. Hirota. Quantum detection and mutual information for qam and psk signals. *IEEE Transactions on Communications*, 47(2):248–254, 1999.

[KPM+22] Bobak Toussi Kiani, Giacomo De Palma, Milad Marvian, Zi-Wen Liu, and Seth Lloyd. Learning quantum data with the quantum earth mover's distance. *Quantum Science and Technology*, 7(4):045002, jul 2022.

[KSL08] Robert L. Kosut, Alireza Shabani, and Daniel A. Lidar. Robust quantum error correction via convex optimization. *Phys. Rev. Lett.*, 100:020502, Jan 2008.

[KSVZ23] Raphael Kaubruegger, Athreya Shankar, Denis V. Vasilyev, and Peter Zoller. Optimal and variational multiparameter quantum metrology and vector-field sensing. *PRX Quantum*, 4:020333, Jun 2023.

[KVS+21] Raphael Kaubruegger, Denis V. Vasilyev, Marius Schulte, Klemens Hammerer, and Peter Zoller. Quantum Variational Optimization of Ramsey Interferometry and Atomic Clocks. *Physical Review X*, 11(4):041045, December 2021.

[KYG+07] Jens Koch, Terri M. Yu, Jay Gambetta, A. A. Houck, D. I. Schuster, J. Majer, Alexandre Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf. Charge-insensitive qubit design derived from the cooper pair box. *Physical Review A*, 76(4):042319, oct 2007.

[LBD15] Tiancheng Li, Miodrag Bolic, and Petar M. Djuric. Resampling Methods for Particle Filtering: Classification, implementation, and strategies. *IEEE Signal Processing Magazine*, 32(3):70–86, May 2015.

[LC98] Jun S. Liu and Rong Chen. Sequential Monte Carlo Methods for Dynamic Systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

[LCL+20] Genyue Liu, Mo Chen, Yi-Xiang Liu, David Layden, and Paola Cappellaro. Repetitive readout enhanced by machine learning. *Machine Learning: Science and Technology*, 1(1):015003, March 2020.

[LCT14] Hoi-Kwong Lo, Marcos Curty, and Kiyoshi Tamaki. Secure quantum key distribution. *Nature Photonics*, 8(8):595–604, July 2014.

[Llo08] Seth Lloyd. Enhanced sensitivity of photodetection via quantum illumination. *Science*, 321(5895):1463–1465, 2008.

[LLSK22] Jonathan Wei Zhong Lau, Kian Hwee Lim, Harshank Shrotriya, and Leong Chuan Kwek. Nisq computing: where are we and where do we go? *AAPPS Bulletin*, 32:27, 09 2022.

[LPGM13] Cosmo Lupo, Stefano Pirandola, Vittorio Giovannetti, and Stefano Mancini. Quantum reading capacity under thermal and correlated noise. *Phys. Rev. A*, 87:062310, Jun 2013.

[LY17] Jing Liu and Haidong Yuan. Quantum parameter estimation with optimal control. *Physical Review A*, 96(1):012117, July 2017.

[LZC+22] Jing Liu, Mao Zhang, Hongzhen Chen, Lingna Wang, and Haidong Yuan. Optimal Scheme for Quantum Metrology. *Advanced Quantum Technologies*, 5(1):2100080, January 2022.

[Man20] Antonio Manzalini. Quantum communications in future networks and services. *Quantum Reports*, 2(1):221–232, March 2020.

[Mar21a] Florian Marquardt. Machine learning and quantum devices. *SciPost Physics Lecture Notes*, page 29, May 2021.

[Mar21b] Florian Marquardt. Online Course: Advanced Machine Learning for Physics, Science, and Artificial Scientific Discovery, 2021.

[Maz10] Jeronimo Maze. *Quantum manipulation of nitrogen-vacancy centers in diamond: From basic properties to applications*. PhD thesis, 01 2010.

[MBD21] Robert D. McMichael, Sean M. Blakley, and Sergey Dushenko. Optbayesexpt: Sequential Bayesian Experiment Design for Adaptive Measurements. *Journal of Research of the National Institute of Standards and Technology*, 126:126002, February 2021.

[MBE21] Johannes Jakob Meyer, Johannes Borregaard, and Jens Eisert. A variational toolbox for quantum multi-parameter estimation. *npj Quantum Information*, 7(1):1–5, June 2021.

[MBS+18] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 11 2018.

[MDB21] Robert D. McMichael, Sergey Dushenko, and Sean M. Blakley. Sequential Bayesian experiment design for adaptive Ramsey sequence measurements. *Journal of Applied Physics*, 130(14):144401, October 2021.

[MFP+22] Christian D. Marciniak, Thomas Feldker, Ivan Pogorelov, Raphael Kaubruegger, Denis V. Vasilyev, Rick van Bijnen, Philipp Schindler, Peter Zoller, Rainer Blatt, and Thomas Monz. Optimal metrology with programmable quantum sensors. *Nature*, 603(7902):604–609, March 2022.

[MGZ+21] Ziqi Ma, Pranav Gokhale, Tian-Xing Zheng, Sisi Zhou, Xiaofei Yu, Liang Jiang, Peter Maurer, and Frederic T. Chong. Adaptive Circuit Learning for Quantum Metrology. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 419–430, October 2021.

[MKH20] Xiao Ma, Peter Karkus, and David Hsu. Particle Filter Recurrent Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:5101–5108, April 2020.

[MM15] C R Müller and Ch Marquardt. A robust quantum receiver for phase shift keyed signals. *New Journal of Physics*, 17(3):032003, mar 2015.

[Mon81] Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais*. Meémoires de l'Acadeémie royale des sciences de Paris vol 1781, 1781.

[MPO21] Filip B. Maciejewski, Zbigniew Puchała, and Michał Oszmaniec. Exploring quantum average-case distances: proofs, properties, and examples. *arXiv:2112.14284*, 2021.

[NBSN19] Murphy Yuezhen Niu, Sergio Boixo, Vadim N. Smelyanskiy, and Hartmut Neven. Universal quantum control through deep reinforcement learning. *npj Quantum Information*, 5(1):33, December 2019.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[NK18] Min Namkung and Younghun Kwon. Sequential state discrimination of coherent states. *Scientific Reports*, 8(1), November 2018.

[NMD12] N. M. Nusran, M. Ummal Momeen, and M. V. Gurudev Dutt. High-dynamic-range magnetometry with a single electronic spin in diamond. *Nature Nanotechnology*, 7(2):109–113, February 2012.

[NOL+21] V. Nguyen, S. B. Orbell, D. T. Lennon, H. Moon, F. Vigneau, L. C. Camenzind, L. Yu, D. M. Zumbühl, G. A. D. Briggs, M. A. Osborne, D. Sejdinovic, and N. Ares. Deep reinforcement learning for efficient measurement of quantum devices. *npj Quantum Information*, 7(1):100, December 2021.

[NPS21] Samuel P. Nolan, Luca Pezzè, and Augusto Smerzi. Frequentist parameter estimation with supervised learning. *AVS Quantum Science*, 3(3):034401, September 2021.

[NSP21] Samuel Nolan, Augusto Smerzi, and Luca Pezzè. A machine learning approach to Bayesian parameter estimation. *npj Quantum Information*, 7(1):169, December 2021.

[NW06] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, 2e edition, 2006.

[ORC+22] Nimba Oshnik, Phila Rembold, Tommaso Calarco, Simone Montangero, Elke Neu, and Matthias M. Müller. Robust magnetometry with single nitrogen-vacancy centers via two-step optimization. *Physical Review A*, 106(1):013107, July 2022.

[ORFW23] Emilio Onorati, Cambyse Rouzé, Daniel Stilck França, and James D. Watson. Efficient learning of ground & thermal states within phases of matter. *arXiv:2301.12946*, 2023.

[Orn73] Donald S Ornstein. An application of ergodic theory to probability theory. *The Annals of Probability*, 1(1):43–58, 1973.

[PAB+20] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani, J. L. Pereira, M. Razavi, J. Shamsul Shaari, M. Tomamichel, V. C. Usenko, G. Vallone, P. Villoresi, and P. Wallden. Advances in quantum cryptography. *Adv. Opt. Photon.*, 12(4):1012–1236, Dec 2020.

[PBJ+15] Michael J. Peterer, Samuel J. Bader, Xiaoyue Jin, Fei Yan, Archana Kamal, Theodore J. Gudmundsen, Peter J. Leek, Terry P. Orlando, William D. Oliver, and Simon Gustavsson. Coherence and decay of higher energy levels of a superconducting transmon qubit. *Phys. Rev. Lett.*, 114:010501, Jan 2015.

[PC19] Gabriel Peyré and Marco Cuturi. Computational Optimal Transport: With Applications to Data Science. *Founda-*

tions and Trends® in Machine Learning, 11(5-6):355–607, 2019.

[PEHM22] Riccardo Porotti, Antoine Essig, Benjamin Huard, and Florian Marquardt. Deep Reinforcement Learning for Quantum State Preparation with Weak Nonlinear Measurements. *Quantum*, 6:747, June 2022.

[Pir11] Stefano Pirandola. Quantum reading of a classical digital memory. *Phys. Rev. Lett.*, 106:090504, Mar 2011.

[PKB+20] Adriano Macarone Palmieri, Egor Kovlakov, Federico Bianchi, Dmitry Yudin, Stanislav Straupe, Jacob D. Biamonte, and Sergei Kulik. Experimental neural network enhanced quantum tomography. *npj Quantum Information*, 6(1):20, December 2020.

[PL14] Nicola Dalla Pozza and Nicola Laurenti. Adaptive discrimination scheme for quantum pulse-position-modulation signals. *Physical Review A*, 89(1):012339, January 2014.

[PLG+11] Stefano Pirandola, Cosmo Lupo, Vittorio Giovannetti, Stefano Mancini, and Samuel L Braunstein. Quantum reading capacity. *New Journal of Physics*, 13(11):113012, nov 2011.

[PLLP19] Stefano Pirandola, Riccardo Laurenza, Cosmo Lupo, and Jason L. Pereira. Fundamental limits to quantum channel discrimination. *npj Quantum Information*, 5(1):50, Jun 2019.

[PPM23] Riccardo Porotti, Vittorio Peano, and Florian Marquardt. Gradient-Ascent Pulse Engineering with Feedback. *PRX Quantum*, 4(3):030305, July 2023. Publisher: American Physical Society.

[Pre99] John Preskill. Notes of the course of quantum information. http://theory.caltech.edu/~preskill/ph229/, 1999. Accessed: 2024-07-01.

[Pre12] John Preskill. Quantum computing and the entanglement frontier, 2012.

[Pre18] John Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, aug 2018.

[QFN21] Yihui Quek, Stanislav Fort, and Hui Khoon Ng. Adaptive quantum state tomography with neural networks. *npj Quantum Information*, 7(1):105, December 2021.

[QZHL22] Yuxiang Qiu, Min Zhuang, Jiahao Huang, and Chaohong Lee. Efficient and robust entanglement generation with deep reinforcement learning for quantum metrology. *New Journal of Physics*, 24(8):083011, August 2022.

[RCLO18] Ernest K Ryu, Yongxin Chen, Wuchen Li, and Stanley Osher. Vector and matrix optimal mass transport: theory, algorithm, and applications. *SIAM Journal on Scientific Computing*, 40(5):A3675–A3698, 2018.

[RD19] Cambyse Rouzé and Nilanjana Datta. Concentration of quantum states from quantum functional and transportation cost inequalities. *Journal of Mathematical Physics*, 60(1):012202, 2019.

[RF21] Cambyse Rouzé and Daniel Stilck França. Learning quantum many-body systems from a few copies. *arXiv:2107.03333*, 2021.

[ROM+20] Phila Rembold, Nimba Oshnik, Matthias M. Müller, Simone Montangero, Tommaso Calarco, and Elke Neu. Introduction to quantum optimal control for quantum sensing with nitrogen-vacancy centers in diamond. *AVS Quantum Science*, 2(2):024701, June 2020.

[SBCMnT10] G. Sentís, E. Bagan, J. Calsamiglia, and R. Muñoz Tapia. Multicopy programmable discrimination of general qubit states. *Phys. Rev. A*, 82:042312, Oct 2010.

[SBCMnT13] G. Sentís, E. Bagan, J. Calsamiglia, and R. Muñoz Tapia. Programmable discrimination with an error margin. *Phys. Rev. A*, 88:052304, Nov 2013.

[SBG+19] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3):032331, mar 2019.

[SC02] Masahide Sasaki and Alberto Carlini. Quantum learning and universal quantum matching machine. *Phys. Rev. A*, 66:022303, Aug 2002.

[SCMTB12] G. Sentís, J. Calsamiglia, R. Muñoz-Tapia, and E. Bagan. Quantum learning without quantum memory. *Scientific Reports*, 2(1), October 2012.

[Ser17] Alessio Serafini. *Quantum Continuous Variables*. CRC Press, 2017.

[SFB20] Jonas Schuff, Lukas J Fiderer, and Daniel Braun. Improving the dynamics of quantum sensors with reinforcement learning. *New Journal of Physics*, 22(3):035001, March 2020.

[SGA15a] Gael Sentís, Mădălin Guţă, and Gerardo Adesso. Quantum learning of coherent states. *EPJ Quantum Technology*, 2(1), July 2015.

[SGA15b] Gael Sentís, Mădălin Guţă, and Gerardo Adesso. Quantum learning of coherent states. *EPJ Quantum Technology*, 2(1):17, December 2015.

[SGK+19] R. Santagati, A. A. Gentile, S. Knauer, S. Schmitt, S. Paesani, C. Granade, N. Wiebe, C. Osterkamp, L. P. McGuinness, J. Wang, M. G. Thompson, J. G. Rarity, F. Jelezko, and A. Laing. Magnetic-Field Learning Using a Single Electronic Spin in Diamond with One-Photon Readout at Room Temperature. *Physical Review X*, 9(2):021019, April 2019.

[SGL+21] Simon Schmitt, Tuvia Gefen, Daniel Louzon, Christian Osterkamp, Nicolas Staudenmaier, Johannes Lang, Matthew Markham, Alex Retzker, Liam P. McGuinness, and Fedor Jelezko. Optimal frequency measurements with quantum probes. *npj Quantum Information*, 7(1):55, December 2021.

[SM23] Leopoldo Sarra and Florian Marquardt. Deep Bayesian Experimental Design for Quantum Many-Body Systems, June 2023.

[SMSM99] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems*, volume 12, 1999.

[SSP14] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. Quantum computing for pattern classification. In Duc-Nghia Pham and Seong-Bae Park, editors, *PRICAI 2014: Trends in Artificial Intelligence*, pages 208–220, Cham, 2014. Springer International Publishing.

[SW21] Adam Scibior and Frank Wood. Differentiable Particle Filtering without Modifying the Forward Pass, October 2021. arXiv:2106.10314 [cs, stat].

[SZBH08] Michal Sedlák, Mário Ziman, Vladimír Bužek, and Mark Hillery. Unambiguous comparison of ensembles of quantum states. *Physical Review A*, 77(4):042304, 2008.

[SZBH09] Michal Sedlák, Mário Ziman, Vladimír Bužek, and Mark Hillery. Unambiguous identification of coherent states. II. Multiple resources. *Physical Review A - Atomic, Molecular, and Optical Physics*, 79(6):062305, jun 2009.

[SZPcv+07] Michal Sedlák, Mário Ziman, Ond řej Přibyla, Vladimír Bužek, and Mark Hillery. Unambiguous identification of coherent states: Searching a quantum database. *Physical Review A - Atomic, Molecular, and Optical Physics*, 76(2):022326, aug 2007.

[TCC+21] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: a review of methods and best practices, 2021.

[TEG+08] Si-Hui Tan, Baris I. Erkmen, Vittorio Giovannetti, Saikat Guha, Seth Lloyd, Lorenzo Maccone, Stefano Pirandola, and Jeffrey H. Shapiro. Quantum illumination with gaussian states. *Phys. Rev. Lett.*, 101:253601, Dec 2008.

[TIO+22] Moeta Tsukamoto, Shuji Ito, Kensuke Ogawa, Yuto Ashida, Kento Sasaki, and Kensuke Kobayashi. Machine-learning-enhanced quantum sensors for accurate magnetic field imaging. *Scientific Reports*, 12(1):13942, September 2022.

[TKL10] Soraya Taghavi, Robert L. Kosut, and Daniel A. Lidar. Channel-optimized quantum error correction. *IEEE Transactions on Information Theory*, 56(3):1461–1473, 2010.

[TSvLL05a] Masahiro Takeoka, Masahide Sasaki, Peter van Loock, and Norbert Lütkenhaus. Implementation of projective measurements with linear optics and continuous photon counting. *Phys. Rev. A*, 71:022318, Feb 2005.

[TSvLL05b] Masahiro Takeoka, Masahide Sasaki, Peter van Loock, and Norbert Lütkenhaus. Implementation of projective measurements with linear optics and continuous photon counting. *Physical Review A*, 71(2):022318, February 2005.

[VDPS23] Seyed Shakib Vedaie, Archismita Dalal, Eduardo J. Páez, and Barry C. Sanders. Framework for Learning and Control in the Classical and Quantum Domains, July 2023.

[Ver13] A. M. Vershik. Long history of the monge-kantorovich transportation problem. *The Mathematical Intelligencer*, 35:1–9, 12 2013.

[Vil08] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[Vin89] Ernest B Vinberg. *Linear representations of groups*. Boston: Birkhauser Verlag, 1989.

[VVH21] Tan Van Vu and Yoshihiko Hasegawa. Geometrical Bounds of the Irreversibility in Markovian Systems. *Phys. Rev. Lett.*, 126:010601, Jan 2021.

[VVS22] Tan Van Vu and Keiji Saito. Thermodynamic Unification of Optimal Transport: Thermodynamic Uncertainty Relation, Minimum Dissipation, and Thermodynamic Speed Limits. *arXiv preprint arXiv:2206.02684*, 2022.

[WGFC14] Nathan Wiebe, Christopher Granade, Christopher Ferrie, and D. G. Cory. Hamiltonian Learning and Certification Using Quantum Resources. *Physical Review Letters*, 112(19):190501, May 2014.

[Wir22] Melchior Wirth. A dual formula for the noncommutative transport distance. *Journal of Statistical Physics*, 187(2):1–18, 2022.

[Wit14] Peter Wittek. 12 - quantum classification. In Peter Wittek, editor, *Quantum Machine Learning*, pages 119–123. Academic Press, Boston, 2014.

[WM08] D.F. Walls and Gerard J. Milburn. *Quantum Optics*. Springer, 2008.

[WPS+17] Jianwei Wang, Stefano Paesani, Raffaele Santagati, Sebastian Knauer, Antonio A. Gentile, Nathan Wiebe, Maurangelo Petruzzella, Jeremy L. O'Brien, John G. Rarity, Anthony Laing, and Mark G. Thompson. Experimental quantum Hamiltonian learning. *Nature Physics*, 13(6):551–555, June 2017.

[WT01] Lex Weaver and Nigel Tao. The optimal reward baseline for gradient-based reinforcement learning. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, UAI'01, pages 538–545, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[XFZ22] Tailong Xiao, Jianping Fan, and Guihua Zeng. Parameter estimation in quantum sensing based on deep reinforcement learning. *npj Quantum Information*, 8(1):1–12, January 2022.

[XLL+19] Han Xu, Junning Li, Liqiang Liu, Yu Wang, Haidong Yuan, and Xin Wang. Generalizable control for quantum parameter estimation through reinforcement learning. *npj Quantum Information*, 5(1):1–8, October 2019.

[XWYW21] Han Xu, Lingna Wang, Haidong Yuan, and Xin Wang. Generalizable control for multiparameter quantum metrology. *Physical Review A*, 103(4):042615, April 2021.

[YKL75] H. Yuen, R. Kennedy, and M. Lax. Optimum testing of multiple hypotheses in quantum detection theory. *IEEE Transactions on Information Theory*, 21(2):125–134, March 1975.

[YPC+22] Jing Yang, Shengshi Pang, Zekai Chen, Andrew N. Jordan, and Adolfo del Campo. Variational principle for optimal quantum controls in quantum metrology. *Phys. Rev. Lett.*, 128:160505, Apr 2022.

[ZDPK+23] Fabio Zoratti, Giacomo De Palma, Bobak Kiani, Quynh T. Nguyen, Milad Marvian, Seth Lloyd, and Vittorio Giovannetti. Improving the speed of variational quantum algorithms for quantum error correction. *Phys. Rev. A*, 108:022611, Aug 2023.

[Zho14] Tao Zhou. Success probabilities for universal unambiguous discriminators between unknown pure states. *Phys. Rev. A*, 89:014301, Jan 2014.

[ZHR+23] I Zohar, B Haylock, Y Romach, M J Arshad, N Halay, N Drucker, R Stöhr, A Denisenko, Y Cohen, C Bonato, and A Finkler. Real-time frequency estimation of a qubit without single-shot-readout. *Quantum Science and Technology*, 8(3):035017, July 2023.

[ZMJ20] Michael Zhu, Kevin Murphy, and Rico Jonschkowski. Towards Differentiable Resampling, April 2020. arXiv:2004.11938 [cs, stat].

[ZPFG21] Fabio Zoratti, Nicola Dalla Pozza, Marco Fanizza, and Vittorio Giovannetti. An agnostic-Dolinar receiver for coherent states classification. *Physical Review A*, 104(4):042606, October 2021.

[ŻS98] Karol Życzkowski and Wojeciech Slomczynski. The Monge distance between quantum states. *Journal of Physics A: Mathematical and General*, 31(45):9095, 1998.

[ŻS01] Karol Życzkowski and Wojciech Slomczynski. The Monge metric on the sphere and geometry of quantum states. *Journal of Physics A: Mathematical and General*, 34(34):6689, 2001.

[ZWA+19] Xiao-Ming Zhang, Zezhu Wei, Raza Asad, Xu-Chen Yang, and Xin Wang. When does reinforcement learning stand out in quantum control? A comparative study on state preparation. *npj Quantum Information*, 5(1):85, December 2019.

[ZYQ06] Chi Zhang, Mingsheng Ying, and Bo Qiao. Universal programmable devices for unambiguous discrimination. *Physical Review A - Atomic, Molecular, and Optical Physics*, 74(4):042308, oct 2006.

[ZYY+22] Mao Zhang, Huai-Ming Yu, Haidong Yuan, Xiaoguang Wang, Rafał Demkowicz-Dobrzański, and Jing Liu. QuanEstimation: An open-source toolkit for quantum parameter estimation. *Physical Review Research*, 4(4):043057, October 2022.