Quantum Inference Protocol : from origination to application

Ho, Matthew Shu Hui

2021

Ho, M. S. H. (2021). Quantum Inference Protocol : from origination to application. Doctoral thesis, Nanyang Technological University, Singapore. <https://hdl.handle.net/10356/153399>

<https://hdl.handle.net/10356/153399>

<https://doi.org/10.32657/10356/153399>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 20:50:12 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

Quantum Inference Protocol

From origination to application

Ho Shu Hui, Matthew

School of Physical & Mathematical Sciences
Division of Physics and Applied Physics

2021

Quantum Inference Protocol

From origination to application

Ho Shu Hui, Matthew

School of Physical & Mathematical Sciences
Division of Physics and Applied Physics

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy in Physics

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research done by me except where otherwise stated in this thesis. The thesis work has not been submitted for a degree or professional qualification to any other university or institution. I declare that this thesis is written by myself and is free of plagiarism and of sufficient grammatical clarity to be examined. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

2 June 2021

.....

Date

.....

Ho Shu Hui, Matthew

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it of sufficient grammatical clarity to be examined. To the best of my knowledge, the thesis is free of plagiarism and the research and writing are those of the candidate's except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

2 June 2021

.....

Date

.....

Asst. Prof. Mile Gu

Authorship Attribution Statement

This thesis contains material from 1 publication in a peer-reviewed journal and 2 publications submitted to a peer-reviewed journals and uploaded on the arXiv. For all 3 manuscripts, I am the first author.

Chapter 4 is published as Matthew Ho, Mile Gu, and Thomas J. Elliott. Robust inference of memory structure for efficient quantum modeling of stochastic processes. Physical Review A **101**, 032327 (2020). DOI: <https://doi.org/10.1103/PhysRevA.101.032327>. The code for the Quantum Inference Protocol is uploaded at <https://github.com/matthew0021/quantum-inference-protocol>.

The contributions of the co-authors are as follows:

- Dr. Thomas J. Elliott provided the initial project direction.
- I carried out all the research work which includes writing the codes for the Quantum Inference Protocol, calculating the analytical derivations of the error scalings, with guidance from Dr. Thomas J. Elliott.
- I prepared the manuscript drafts which were later edited by Asst. Prof. Mile Gu and polished by Dr. Thomas J. Elliott.

Reprinted figure with permission from Matthew Ho, Mile Gu, and Thomas J. Elliott, Physical Review A 101, 032327 (2020). Copyright 2020 by the American Physical Society.

Chapters 5 and **6** constitutes one manuscript can be found at the arXiv link <https://arxiv.org/abs/2105.06448>. The work was done with Ryuji Takagi and Asst. Prof. Mile Gu. The codes for unitary construction and decomposition can be found at <https://github.com/matthew0021/unitary-construction-and-decomposition>.

The contributions of the co-authors are as follows:

- I provided the initial project direction.
- Dr. Ryuji Takagi and I held many discussions on the theory and notation behind quantum error mitigation for application on near term quantum computers.

-
- All codes were written by me. The codes are:
 - (1) Constructing the unitary operator.
 - (2) Decomposing an arbitrary 2^n by 2^n unitary operator into elementary quantum gates.
 - (3) Error mitigation: pre-experimental codes which include numerical work on IBM's Qiskit for gate set tomography.
 - (4) Error mitigation: post processing of gate set tomography data.
 - (5) Error mitigation: post-experimental codes for post-processing the Monte Carlo data.
 - I calculated the analytical derivations for finite-shots for gate set tomography, δ -tolerance parameter, and scaling of fidelity.
 - I prepared the manuscript drafts which were refined after discussion with Dr. Ryuji Takagi and Asst. Prof. Mile Gu.

Chapter 7 can be found at the arXiv link <https://arxiv.org/abs/2103.14053>. The work was a collaboration between Andri Pradana, Thomas J. Elliott, Assoc. Prof. Chew Lock Yue, and Asst. Prof. Mile Gu.

The contributions of the co-authors are as follows:

- I provided the initial project direction.
- Mr. Andri Pradana provided the numerical analysis of the classical statistical complexity.
- Assoc. Prof. Chew Lock Yue and Mr. Andri Pradana provided valuable insights from the classical causal modelling perspective.
- Mr. Andri Pradana, Dr. Thomas J. Elliott, Assoc. Prof. Chew Lock Yue, and Asst. Prof. Mile Gu participated and engaged in discussions for interpreting and consolidating the results.
- I prepared the drafts, Asst. Prof. Mile Gu edited and refined the drafts, and Dr. Thomas J. Elliott polished the final manuscript.

2 June 2021

.....
Date

.....
Ho Shu Hui, Matthew

Abstract

We are at the interface of two hot research areas – modelling of stochastic processes with causal models in quantitative science, and quantum simulation on real quantum devices. Classical causal models first made its appearance in 1989 with the aim of extracting a stochastic process’ underlying dynamics in order to do statistically faithful predictions. Research into causal models expanded into the quantum regime where it was shown that quantum models could outperform classical models; quantum models had equal predictive capabilities but had less memory requirement. This kickstarted the exploratory research into building more optimal quantum models with lower memory requirements. However, all quantum models required prior knowledge of classical causal models. Errors that occurred in classical models due to finite-length stochastic processes were automatically inherited by quantum models.

The Quantum Inference Protocol was developed to circumvent the problem of inheriting errors from classical causal models by bypassing the requirement of using classical causal models to build quantum models. These quantum models feature probabilities that are inferred through frequency counting. In Chapter 4, we demonstrate the equivalence between inferred quantum models from the Quantum Inference Protocol and exact quantum models obtained from classical models. We provide analytical reasoning and two examples to validate and justify the efficacy of the Quantum Inference Protocol.

With the increasing popularity of open source cloud quantum computing, Chapter 5 introduces an algorithm to reconstruct the relevant unitary operator for the inferred quantum model. Additionally, we provide a separate algorithm to decompose the unitary operator into elementary quantum gates that are implementable on real quantum devices. This sees a natural progression to apply the inferred quantum models on a fully functioning quantum device, since previous works focused on exploratory work in the ideal noiseless scenario. Due to hardware limitations, quantum devices are inherently noisy, reducing the accuracy of the measurement outcomes. We seek to apply error mitigation techniques to reduce the effects of noise from quantum devices. In Chapter 6, we outline the full procedure for an error mitigation method known as probabilistic cancellation method before conducting simulations with noise models from real quantum backends. Our results indicate that error mitigation aids in suppressing the effects of noise for statistical distributions.

On second side of the same coin, we apply the Quantum Inference Protocol to extract the dynamics of elementary cellular automata in Chapter [7](#). We show that our inferred quantum models are able to identify varying degrees of complexity in cellular automata, offering a fresh perspective in the form of a hybrid classification scheme.

Acknowledgements

First and foremost, a disclaimer. There are many people I wish to thank but there is a word limit on this thesis. Yes, this includes the acknowledgements. It's a pity. I could have written a novel. Anyway, here goes.

My deepest gratitude to my supervisor Asst. Prof. Mile Gu and co-supervisor Assoc. Prof. Chew Lock Yue.

To Mile, your creativity and ability to simplify the most complex concepts has inspired me greatly. Thank you for graciously granting the countless opportunities for engagement with other colleagues in the world. You have definitely helped to broaden my perspectives by—without any hints of hesitation—sponsoring trips to conferences and workshops in Madrid, Korea, and Shenzhen when travel was possible.

To Lock Yue, thank you for the inspiration and advice on research while walking with me, providing advice in life and in the PhD. Your passion for understanding the fundamentals of science has impacted me positively and I am thankful and grateful for your supervision.

To Thomas, or more personally known as Tom, I count myself extremely lucky to have started the PhD journey under your mentorship. Thank you for guiding me in the first half of my PhD stint with patience and constructive criticism. I'm glad we stuck to weekly discussions from the beginning that saw us tearing down my arguments and shaping my thought processes. I have learnt so much. I'll never forget our trip to Korea for a conference and the many conversations about coffee, tea, cats, and life!

To Jeremiah, a beloved brother whom I have known for the past two decades. Bro, I could not have asked for a better person to celebrate joyful times and go through rough times with. You've seen me through this PhD journey, listening to my gripes and struggles, giving life advice along the way. You always inspire me to be a better person every day. I can't believe we have known each other for two decades. I'm excited for what the future holds for our friendship as we venture forward onto different life stages. I won't forget our special *bromantic* moments we had vacuuming your car from the suspected roach infestation or enjoying the sunset with Kim and

Kat back in mid-2020!

To Huan, biggest appreciation for walking with me through fun times and providing counsel when the path was seemingly very narrow. I owe you a long-overdue catch up session over prata and tea!

Special mention to a few people. Wei Liang, a big brother and fellow musician, bassist, and life advisor, thank you for the many great times and for proof-reading this entire thesis; Guo Yao and Minjeong, my colleagues and badminton buddies who keep me striving for excellence on the badminton court, I will definitely miss badminton with you guys; special thanks to Minjeong for meticulously scrutinising the details of this thesis as well; Grayson, my fellow cross-island travel buddy and PhD colleague that I rant to during our cross-island journeys, whom I've known since undergraduate days, travelling to Sweden with and eating burnt cooking countless times; Ryuji, for the inspiration and enthusiasm you exude; and Josh, for being a very encouraging friend and confidant; thank you all!

To the Nonlinear and Complex Systems Lab mates, past and present, Suryadi, Andri Pradana, Wei Liang, Ayana Aspembitova, Khi Pin, He Chong, Ritvik Dobriyal, Luca Vismara, Saw Vee Liem, He Lian Jie, thank you! To the Complexity Institute / Quantum Hub colleagues, past and present, Suen Whei Yeap, Felix Binder, Andrew Garner, Alec Boyd, Paul Riechers, Jayne Thompson, Varun Narasimhachar, Liu Qing, Yang Chengran, Alex Wozniakowski, Bartosz Regula, Guo Yao, Minjeong, Ximing, Ryuji Takagi, Keith Ng, Spiros Kechrimparis, Xiao Yunlong, thank you!

And no, I didn't forget my family. To my family, Mum, Dad, and not-so-little-anymore-Sister, your steadfast love, supporting me through the highs and lows, tolerating my idiosyncrasies, loving me unconditionally whether or not I drop out from the PhD programme or not, has given me great strength to persevere on. I could not have asked for a more understanding family and I hope I have made you proud with this piece of work (this thesis that you'll probably not read).

Not forgetting my better half, Shelley. You've truly been my pillar of support and grounding me through tough times. Thank you for loving me for who I am and teaching me how to love. I could never have asked for a better life partner than you and I'm ever so thankful for you. The end of this PhD marks the beginning of many hurdles ahead but I am sure heading into the future together will most certainly be colourful, in faith, hope, and love.

Last of all, I give thanks to Almighty God for His providence and infinite grace that saw me through this PhD journey. I count myself extremely privileged to be able given this opportunity—and I quote from the late Jacob Bekenstein—to *look at the world as a product of God, He set very specific laws and we delight in discovering them through scientific work.*

Contents

1	Introduction	1
2	Preliminaries	7
2.1	Classical	7
2.1.1	Probability theory	7
2.1.2	Entropy	8
2.2	Quantum	10
2.2.1	States	10
2.2.2	Operators	11
2.2.3	Measurements	12
2.2.4	Entropy	14
2.2.5	Pauli transfer matrix	14
2.3	Stochastic Processes	15
3	Causal models	17
3.1	Classical causal models	18
3.1.1	How to build an ϵ -machine	21
3.1.2	C_μ vs mutual information	24
3.2	Quantum causal models	26
3.2.1	Version 1: Kraus operations (2012)	26
3.2.2	Version 2: Cryptic order (2016)	32
3.2.3	Version 3: Unitary quantum causal models (2018 onwards)	39
4	Quantum Inference Protocol	42
4.1	Motivation	42
4.2	The set-up	44
4.3	Self-merging at and above the Markov order	46
4.4	Robustness of C_q	48
4.5	The Inference Protocol	51
4.6	Scaling of the error	53
4.7	Examples	55
4.8	Summary	57

5	Unitary construction and decomposition	59
5.1	Merging quantum memory states	59
5.2	Building the unitary	60
5.3	Decomposing the unitary	62
5.4	δ -similarity of quantum memory states	66
6	Quantum Error Mitigation	70
6.1	Methodology for error mitigation	71
6.1.1	Gate set tomography.	72
6.1.2	Quasiprobability decomposition.	73
6.1.3	Monte Carlo simulation.	76
6.1.4	Post-processing results.	77
6.2	Results	80
6.2.1	Multi-step error mitigation	81
6.2.2	Individual-step error mitigation	84
6.2.3	Memory costs	86
6.2.4	Scaling for finite number of GST shots	88
6.2.5	Scaling of fidelity	91
7	Classifying Elementary Cellular Automata	94
7.1	Cellular Automata	95
7.2	Classifying Cellular Automata	97
7.3	A stochastic perspective	101
7.4	Complexity criterion	101
7.5	The truncated methodology	102
7.6	Numerical Results	104
7.7	Discussion	108
8	Conclusion and future works	110
A	ECA Methodology	118
A.1	Methodology for ECA	118
A.2	Boundary conditions	120
A.3	Longer L , larger $t_{\max}$	121

List of Figures

2.1	A concise representation of how the Shannon entropies of various random variables are related.	9
2.2	The Bloch sphere. The red/blue/green points represent (eigen)states of the Pauli X/Y/Z operators. An arbitrary qubit (yellow point) can be represented as $ \psi\rangle = \cos\frac{\theta}{2} 0\rangle + e^{i\phi}\sin\frac{\theta}{2} 1\rangle$. ($0 \leq \theta \leq \pi$, $0 \leq \phi \leq 2\pi$). Pure states lie on the surface of the sphere while mixed states reside within the sphere.	12
3.1	Equivalence classes are formed by grouping pasts with statistically identical futures.	19
3.2	An example of an ε -machine. The nodes represent causal states and directed edges indicate the transitions between causal states. Each transition occurs with some probability p and is accompanied with an output x . The statistical complexity of the ε -machine is defined as the Shannon entropy over the stationary distribution of the causal states.	20
3.3	The sub-tree reconstruction algorithm, here illustrated for $L = 3$.	22
3.4	The ε -machine for the perturbed coin process.	28
3.5	The blue (red) line represents how C_q (C_μ) scales with p . Note that at $p = 0.5$, $C_\mu = 0$.	31
3.6	The ε -machine for the 4-3 Golden Mean Process.	33
3.7	The 4-3 Golden Mean Process. Analytical results for $C_q(L)$ with various values of p are plotted up to $L = 5$. The cryptic order is $k = 3$. Observe that $C_q(L)$ plateaus when $L \geq k$.	38
3.8	The red line represents $C_q(L)$ while the blue line represents C_μ , both computed with the 4-3 Golden Mean Process set to $p = 0.2$.	38
3.9	The unitary operator acts on the quantum memory state $ \sigma_j\rangle$ with a blank ancillary state to produce x when measured in the appropriate basis.	39

4.1	The Quantum Inference Protocol analyses stochastic processes directly to build quantum causal models without any reliance on classical inference algorithms. It is expected to have similar memory requirements as the exact quantum causal models, $\tilde{C}_q \approx C_q$. Its relationship to prior quantum causal models is illustrated here.	43
4.2	Repeated applications of the unitary operator on the memory register, now given with $ \varsigma_{x_{-L:0}}\rangle$, with blank ancillary qubits $ 0\rangle$ allows the statistical replication of the process.	45
4.3	(a) 3-2 Golden Mean Process with $p = 0.9$. (b) Trace distance of $(\tilde{P}(X_0 0X_{-L:0})$ and $\tilde{P}(X_0 1X_{-L:0})$ vs L . (c) $\tilde{C}_q(L)$ vs L .	56
4.4	(a) Nemo Process set to $p = 0.1$ and $q = 0.9$. (b) Trace distance of $(\tilde{P}(X_0 0X_{-L:0})$ and $\tilde{P}(X_0 1X_{-L:0})$ vs L . (c) $\tilde{C}_q(L)$ vs L .	56
5.1	The unitary is decomposed into elementary quantum gates comprising of CNOTs, controlled-Z rotations, controlled-Y rotations, controlled-phase gates. We labelled u_1 , u_2 , v_1 , and v_2 to represent the segments according to Eq. (5.7). The multiplexor is made up of diagonal cosine and sine terms (see Eq. (5.9)).	63
5.2	(a) Unitary quantum models operate with a memory qubit $ \sigma_j\rangle$ and an ancillary blank qubit $ 0\rangle$ that will undergo unitary operation to produce $ \sigma_{\lambda(x,j)}\rangle$ and $ x\rangle$. (b) In the quantum circuit picture, q_0 functions as the memory (control) qubit that acts on target ancillary blank qubits q_1 , q_2 , q_3 , and so on. (c) Each unitary in (b) is decomposed in a series of rotations, CNOTs, NOT gates, and the multiplexor gate.	69
6.1	The general workflow for the error mitigation method comprises of 4 main steps.	72
6.2	Initialise each Monte Carlo run with state $\bar{\rho}_k = q_0 q_1\rangle\langle q_0 q_1 $, apply operation $\tilde{O}^{(l)}$ (in this case it is the unitary operator U), apply basis operation $\tilde{B}_i$, and measure in basis $\tilde{M}_j$. Let the measured outcome be μ .	77
6.3	The perturbed coin ε -machine is used as a generator to generate a stochastic process with parameter $p = 0.2$. After using the Quantum Inference Protocol to generate quantum memory states, the unitary operator can be constructed and decomposed. The statistical distribution of outputs can then be obtained with and without error mitigation to compare the efficacy of error mitigation.	80
6.4	The multi-step joint statistical distributions for (a) 1 time step of unitary operation with outputs given by $x_{\text{output}} \equiv x^{(t=1)} \in \{0, 1\}$ and (b) 2 time steps of unitary operator with outputs given by $x_{\text{outputs}} \equiv x^{(t=1)}x^{(t=2)} \in \{00, 01, 10, 11\}$, all of which are obtained with $N_{\text{MC}} = 10^7$ samples.	82

6.5	Distribution for the expectation values for $P_{\text{QEM}}(x^{(t)} = 1)$ i.e. for outcome $x = 1$ for (a) $t=1$ time step of unitary operation, (b) $t=2$ time steps of unitary operations, and (c) $t=3$ time steps of unitary operations. All figures had error mitigation carried out (blue histogram) and the results for the output 1 are compared against the ideal quantum simulator without noise (black) and quantum simulator with noise (green). The histograms are constructed with 10^3 chunks, each chunk having $N_{\text{MC,chunk}} = 10^4$ samples.	85
6.6	The perturbed coin has the following C_μ and C_q for $0 \leq p \leq 1$. C_μ is discontinuous only at $p = 0.5$ as the ε -machine at $p = 0.5$ is essentially a random process that requires only a single causal state. The region where $0.4688 \leq p \leq 0.5134$ ($p \neq 0.5$) denotes the region where the cumulative memory cost for the quantum models require outdoes the memory cost for the ε -machine when error mitigation methods are taken into account.	87
6.7	A simple illustration of how a small number of Monte Carlo samples is more likely to produce a distribution that crosses into ill-defined regions. The vertical axis here represents the frequency of obtaining error mitigated expectation value of x	91
7.1	(a) Each cell's state in elementary cellular automata (ECA) is updated by simple local update rules. These rules are permutations of its past nearest neighbours cells' states and its past state. A rule number is determined by converting the outcomes of states at t from decimal to binary, for example (a) shows ECA Rule $\{00011010\}_{\text{binary}} = \{26\}_{\text{decimal}}$. (b) Four qualitative classes of ECA are observed when the ECA is initialised with random initial conditions.	96
7.2	Rule 18 with and without a filter. The filter indicates the location of the kinks.	100
7.3	Our workflow to obtain the quantum statistical memory $\tilde{C}_q^{(t)}$ of the ECA at each time step. For completeness, we use the classical sub-tree algorithm to infer $\tilde{C}_\mu^{(t)}$ of the ECA at each time step.	103
7.4	A spectrum of ECA complexity with Rule 30 (Class III, simple) and Rule 110 (Class IV, complex) at the extremes. On each plot, the lines represent the average $C_q^{(t)}$ from 5 random initial conditions. Error bars are present but too small to be visible at scale shown. Rules 22, 18, and 122 are Wolfram Class III while Rule 54 is Class IV – our spectrum is consistent with this division, but shows further nuance. Plots for all rules can be found in the Supplementary Material.	105
7.5	Class I and II.	106
7.6	Class III and IV plotted on a continuous spectrum.	107

8.1	Two streams of water meet and form a new stream. The new stream may also branch out into different streams. Replace the word 'stream' with 'scientific topic'. Illustration drawn by the author of this thesis.	111
8.2	The thesis can be summarised with this workflow that traces the origins of the Quantum Inference Protocol as well as how it branches out into various fields of research.	112
A.1	Generation of finite-width ECA evolution with open boundaries via extended ECA with periodic boundaries.	118
A.2	Four rules (each from Wolfram's classes) are used to illustrate that there is little quantitative difference between ECAs with periodic boundary conditions and ECAs with open boundary conditions.	120
A.3	$C_q^{(t)}$ plots for a selection of rules with longer L and larger $t_{\max}$. Plots shown for $\{W = 64,000, L = 6\}$, $\{W = 128,000, L = 7\}$, and $\{W = 256,000, L = 8\}$.	121
A.4	Over time, the finite number of gliders present in the initial configuration of a finite-width Rule 110 ECA tends to disappear due to annihilation with other gliders. At longer times, there are then fewer gliders to propagate longer-range correlations across the ECA.	122

List of Tables

2.1	The different circuits for measuring in each of the three Pauli X , Y , and Z bases. Measurement outcomes are given in logical 0 and logical 1 corresponding to the $+1$ and -1 eigenstates of the bases.	13
6.1	The 13 CPTP maps listed here are quantum channels that can be used for basis operations. id represents the identity channel while $\mathcal{P}_{ \cdot\rangle}$ signifies a state preparation channel that prepares the state $ \cdot\rangle$.	75
6.2	A system of $n = 2$ qubits requires 241 basis operations. The first 169 can be obtained by taking the tensor product of the 13 basis operations listed in Table 6.1 while the remaining 72 basis operations are listed here. Here, we let $\mathcal{K} = \mathcal{S} \circ \mathcal{H}$.	76

CHAPTER 1

Introduction

We live in an era where quantum technologies is anticipated to have a large impact on our lives, so much so that the word “quantum” has become a buzzword of every-day conversation. Television manufacturers advertise using quantum technologies in the form of quantum dots [1]; Samsung has manufactured smartphones that use quantum random number generator for enhanced security purposes [2, 3]; even movies use the word quantum in their title just like the James Bond franchise Quantum of Solace [4]. To the physicist, these may seem like a very superficial use of the word “quantum”, almost misleading at times. But it is clear that the word “quantum” has become a magic word that carries the connotation of something that can emanate an enigmatic power of sorts [5]. Nevertheless, the growing presence of quantum concepts in science and technology has increasingly inspired the research community to find areas where quantum mechanics has clear advantages to advance the scientific frontier. Thus far, there have been numerous advantages that quantum mechanics offers such as in factoring large numbers [6], search algorithms [7], quantum cryptography [8–10] and many more, not to mention

the abundant ideas that branch from these examples.

In the department of complexity science where the study of time series and stochastic processes, models have been built to understand the underlying dynamics of these processes. It is only when the underlying dynamics are understood with greater clarity that these models provide some form of predictive capabilities. Computational mechanics has been a major contender of producing predictive models of stochastic processes [11, 12]. The philosophy that drives the improvement of these predictive models under the framework of computational mechanics is Occam's Razor. It states that one should not need any more information than necessary to predict the future of a given process' past. Any additional information on top of the required minimum is considered redundant and will not serve any more purpose than what is necessary, only consuming additional resources. This philosophy has driven the development of classical causal models (more familiarly known as ϵ -machines) [11-14] which has been applied to study the dripping faucet experiment [15], model stock markets [16], and inferring the minimal model for neural activity [17, 18].

For example, a potential solution to predicting the future of a complete random process would be to use brute force to track all possible pasts and the futures that these pasts lead to. The memory requirement grows indefinitely. However, a completely random process should not need any resources to predict its future as all futures are equally likely. A random process is said to be statistically simple. In the same vein, a completely uniform process (such as a periodic process) should not require large amounts of resources for prediction as well; one only has to remember as far back as a process' periodicity in order to know what the futures will be. In contrast, a complex process consisting of a mix of periodicity and randomness requires more resources for prediction. It is neither periodic nor statistically simple. Instead, it

carries some degree of correlation between each time step. In fact, it was established that the operational perspective of causal modelling is closely linked to our intuition of what “complexity” entails, given in the form of quantifying the amount of resources required by a causal model for predictive capabilities [13].

Recently, it was established that through an appropriate one-to-one mapping of the classical model’s states to non-orthogonal quantum states, quantum models would have an edge over their classical counterparts as the amount of memory resources can be reduced while retaining predictive capabilities [19, 20]. Greater memory reduction can be achieved by increasing the degree of non-orthogonality between quantum states in the quantum model [21, 22]. In fact, the best known quantum models with the greatest memory reduction as of the writing of this thesis are phase-dependent unitary quantum models that have a complex phase factor appended to the quantum states [23, 24]. These unitary quantum models are constructed from the perspective of preparing quantum models for quantum computers. Obtaining the appropriate complex phases is a manual task with no known systematic methodology, but this minor obstacle does not undermine the successes and advantages of phaseless unitary quantum models.

In the real world where stochastic processes are of finite length, probability distributions obtained through frequency counting or some other means will most certainly incur some errors. Computational mechanics hence necessitates the use of probability smoothing techniques to build proper classical causal models. Should the wrong tolerance be used, the resultant classical model will be inaccurate to the point of having an oversimplified model. The inaccuracy is carried over to the analogous quantum models.

Nevertheless, memory advantages still exist for the now-inaccurate quantum models despite being mapped from these inaccurate classical causal models. However, knowing that the quantum models are inaccurate poses a question of practicality. Should quantum models should be used even though there is a high chance of being inaccurate quantum models?

The solution presented itself with the development of the Quantum Inference Protocol. The Quantum Inference Protocol was developed to abolish dependence on classical causal models by providing an independent method of constructing phaseless unitary quantum models without the use of any probability smoothing method [25]. This meant that phaseless unitary quantum models are built directly from the stochastic process itself. These phaseless unitary quantum models are as accurate as the limits of finite-length stochastic processes permit.

With an escalating number of researchers jumping on the quantum bandwagon to develop and advance the quantum frontier through algorithms and hardware, it is expected that the use of quantum technologies will increase alongside. Current quantum computers are presently noisy and have been called Noisy Intermediate-Scale Quantum (NISQ) devices [26]. NISQ devices are inherently noisy, with the noise mostly arising from hardware restrictions. However, these near-term quantum computers are expected to be more advantageous over classical computers [27, 28]. Traditional methods of suppressing the effects of noise were proposed in the form of error correction codes [29–32] as a step towards fault-tolerant quantum computing. Though highly accurate, error correcting codes typically require many qubits just to correct errors in one qubit, limiting its practicability. Alternate approaches to reduce the effects of noise from near-term quantum devices have been developed in the form of quantum error mitigation [33–40], which required far less qubits than

error-correction codes. Error mitigation techniques were introduced by a team at IBM [33] and experimentally verified in a single qubit system [37].

The current climate of open source quantum computing such as the IBM Quantum Experience promotes conducting one's own experiments on a quantum computer. As prior research focused solely on building the best quantum models, we choose to focus on implementation to bridge the divide between theoretical and experimental research. Our objective is to implement and enhance unitary quantum models on the IBM Quantum Experience with error mitigation. Through this, we demonstrate the effectiveness of error mitigation for our quantum models comprising of two qubits [41] as well as derive scaling bounds.

Another area of research within the field of quantitative science that is closely linked to computation is cellular automata. A cellular automaton comprises of a set of cells with states that update every time step based on permutations of past states. These permutations of the past are known as update rules and have the capacity to produce different behaviour in cellular automata. The simplest cellular automata, known as elementary cellular automata (ECA), is a one dimensional chain of cells. Even with its simplicity, ECA rules are able to generate a wide variety of behaviour when initialised with random initial conditions as well as generate fractals with simple initial conditions. These demonstrate the capabilities of simple ECA update rules. Viewed in a different light, the update rules in ECA transforms one stochastic process to another at each time step. Thus the computational mechanics framework can be employed to provide a proper quantification of underlying causal structure within ECA for these update rules. We demonstrate that the Quantum Inference Protocol is able to identify discrete classes alongside ambiguously classified update rules, allowing us to formulate a hybrid of discrete and continuous classification schemes [42].

This leads us to the summary of this thesis. As a whole, this thesis documents what has been done while providing the relevant methodology for future research. We will briefly cover the important notations and relevant information required to understand stochastic processes, quantum information theory, and superoperators in Chapter [2](#). Chapter [3](#) will then introduce the history of classical causal models and the various versions of quantum models, each providing a leap of improvement over its predecessor. The focus of this thesis begins in Chapter [4](#) which is based on our publication in Ref. [\[25\]](#). We demonstrate how quantum models can be built independently of classical causal models with the Quantum Inference Protocol. We also show that the Quantum Inference Protocol has a dual role in quickly identifying the estimated amount of memory required for quantum models to operate without first building the quantum states.

Inferring quantum states using the Quantum Inference Protocol is insufficient for modelling; a unitary operator is required for the model to work. In Chapter [5](#), we will outline the procedure for constructing the unitary operator. Additionally, we introduce a method for decomposing the unitary operator into implementable elementary quantum gates that can be applied to quantum devices. In Chapter [6](#), we step into the realm of superoperator formalism to tackle the task of error mitigation for a system of two qubits with a newly proposed set of basis operations. Using gate set tomography to obtain the inverse noise profile and a Monte Carlo sampling, we show that the error mitigation is able to reduce the effects of noise when simulating on near-term quantum computers [\[41\]](#). We will round off the thesis with an application of the Quantum Inference Protocol in Chapter [7](#). Here, we propose a refined classification scheme for elementary cellular automata based on a hybrid of discrete and continuous classes [\[42\]](#) which the quantum statistical memory can identify.

CHAPTER 2

Preliminaries

Statistics: the mathematical theory of ignorance.

- Morris Kline

In this Chapter, we provide only the necessary preliminary information required to understand this thesis. The reader is encouraged to refer to textbooks for more information on topics which are not covered in this Chapter.

2.1 Classical

2.1.1 Probability theory

Mathematicians represent what is commonly known as chance with probabilities. For example, a six sided fair dice has equal probability of landing on all six sides facing up

and probabilities of all sides must sum to unity. Each trial in a random process such as dice throws can be represented with random variable X that takes on outcome x .

The joint probability distribution denotes the probability of seeing two or more outcomes together. Typically, one has to factor in the scenario of dependence – if the next outcome is dependent on the previous outcome. For example, if X_2 follows X_1 , then the joint probability distribution of X_1 and X_2 is

$$P(X_1, X_2) \equiv P(X_2|X_1)P(X_1). \quad (2.1)$$

Eq. (2.1) is particularly useful to find out the probability of two consecutive dice throws if they were somehow magically dependent on each other—if odd numbers are more likely to follow numbers and vice versa, for instance.

Since unbiased consecutive dice throws have no correlations and are said to be identically and independently distributed (i.i.d.), the joint probability distribution in Eq. (2.1) reduces to

$$P(X_1, X_2) = P(X_1)P(X_2) \quad (2.2)$$

because $P(X_2)$ does not depend on the outcome X_1 .

2.1.2 Entropy

The Shannon entropy quantifies the amount of uncertainty in informational terms. Since we adopt base 2 for the logarithmic term, information is measured in units known as *bits*. For a random variable X , the Shannon entropy is

$$H(X) \equiv - \sum_i P(x_i) \log_2 P(x_i) \quad (2.3)$$

where $P(x_i)$ is the probability of random variable X giving an outcome x_i . As with joint probabilities, the joint entropy indicates the uncertainty of two random variables

Figure 2.1: A concise representation of how the Shannon entropies of various random variables are related.

X and Y . It is given by

$$\begin{aligned} H(X, Y) &\equiv H(Y|X) + H(X) \\ &\equiv H(X|Y) + H(Y). \end{aligned} \tag{2.4}$$

The term $H(Y|X)$ (and likewise for $H(X|Y)$) is a conditional entropy which is similar to the conditional probability. It represents the uncertainty of observing outcome Y given that X has occurred. This allows us to find the mutual information between random variables X and Y , which quantifies the amount of correlations between them.

The mutual information can be written in various ways as such

$$\begin{aligned} I(X; Y) &\equiv H(X) - H(X|Y) \\ &\equiv H(Y) - H(Y|X) \\ &\equiv H(X, Y) - H(X) - H(Y). \end{aligned} \tag{2.5}$$

The relationship between these entropies can be captured in a Venn diagram in Fig.

2.1.

2.2 Quantum

2.2.1 States

The matrix formalism provides a convenient way of visualising discrete, finite dimensional quantum dynamics. For an n -qubit system, a quantum state is given by a ket which is a column vector with elements which may be complex,

$$|\psi\rangle \equiv [c_1, c_2, \dots, c_{2^n}]^T, \quad (2.6)$$

where $\sum_i^{2^n} |c_i|^2 = 1$ for normalisation.

A vector dual to the ket is given by a bra. It is a row vector with elements as the complex conjugate of Eq. [2.6](#),

$$\langle\psi| \equiv [c_1^*, c_2^*, \dots, c_{2^n}^*]. \quad (2.7)$$

Quantum states may have complex phases $e^{i\phi}|\psi\rangle$ where ϕ is a real number. The $e^{i\phi}$ term is known as the global phase and is typically ignored because $e^{i\phi}|\psi\rangle$ and $|\psi\rangle$ produce equivalent measurement statistics. For phaseless $\phi = 0$ quantum states, the inner product $\langle\psi|\psi'\rangle$ is the matrix multiplication between a row vector and a column vector that gives a number between 0 and 1. When the inner product is zero, the two quantum states are orthogonal states. At the other extreme when the inner product is one, the two quantum states are the same state, $\langle\psi|\psi'\rangle = \langle\psi|\psi\rangle = 1$. For any value $0 < \langle\psi|\psi'\rangle < 1$, the two states are non-orthogonal but have some degree of overlap.

A generalised way of describing mixtures of quantum states is the density matrix, ρ . It can be used to describe both pure states and mixed states. Pure states are states that can be represented as a single state vector (row- or column-wise) and has full information about the quantum system and hence $\rho = |\psi\rangle\langle\psi|$. Mixed states are

mixtures of pure states occurring with various probabilities $P(|\psi_i\rangle)$ and hence

$$\rho = \sum_i P(|\psi_i\rangle) |\psi_i\rangle \langle \psi_i|. \quad (2.8)$$

ρ can be expressed in terms of Pauli operators as

$$\rho = \frac{1}{2} (I + r \sin \theta \cos \phi \sigma_x + r \sin \theta \sin \phi \sigma_y + r \cos \theta \sigma_z), \quad (2.9)$$

where

$$\begin{aligned} I &= |0\rangle \langle 0| + |1\rangle \langle 1| = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; & \sigma_x &= |1\rangle \langle 0| + |0\rangle \langle 1| = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \\ \sigma_y &= i|1\rangle \langle 0| - i|0\rangle \langle 1| = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}; & \sigma_z &= |0\rangle \langle 0| - |1\rangle \langle 1| = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \quad (2.10)$$

are the Pauli operators. This allows a qubit to be represented on a sphere with radius $r = 1$ known as the Bloch sphere, given in Fig. [2.2](#).

2.2.2 Operators

A quantum state transforms to another state when an operator acts on it. An n -qubit system requires an n -qubit operator given in the matrix formalism of quantum mechanics as a $2^n \times 2^n$ complex-valued matrix. An operator acting on a ket is written as $A|\psi\rangle$ while an operator acting on its dual space is written as $\langle \phi| A^\dagger$ where $A^\dagger = (A^*)^T$ is the self-adjoint of A . The class of operators that are equal to their self adjoint are Hermitian operators, while operators that have $A^\dagger A = I$ are unitary operators [\[43\]](#).

In quantum circuits, quantum states are transformed by action of unitary operators. This action rotates quantum states on the Bloch sphere. However, a quantum circuit is incomplete if we only have states and operators. One needs to do measurements

Figure 2.2: The Bloch sphere. The red/blue/green points represent (eigen)states of the Pauli X/Y/Z operators. An arbitrary qubit (yellow point) can be represented as $|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle$. ($0 \leq \theta \leq \pi$, $0 \leq \phi \leq 2\pi$). Pure states lie on the surface of the sphere while mixed states reside within the sphere.

to ascertain the outcome of the operators acting on the states.

2.2.3 Measurements

Measurements allow one to learn about the system. In quantum mechanics, measuring a quantum state with operator M , the possible outcomes would be m_n , the eigenstates of M . The state is then collapsed into the one of the eigenvectors of the measurement basis.

Quantum devices typically measure in the computational basis (which is the Pauli Z basis). The Pauli Z matrix has eigenstates $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ with eigenvalues +1 and -1 respectively. Performing a Pauli Z measurement results in a logical zero

or logical one. If the measurement outcome is a logical zero (logical one), then the state is the $+1$ (-1) eigenstate of the Pauli Z measurement.

To perform a Pauli X or Y measurement, one carries out unitary transformations on the Pauli Z basis measurement such that the measurement outcomes are in the ± 1 eigenvectors of the transformed measurement basis. In Table 2.1 we demonstrate the various unitary transformations necessary to measure in Pauli X or Y basis. The unitary transformations require a Hadamard gate and the conjugate of S gate (i.e. $S^\dagger$). The Hadamard and $S^\dagger$ gate are given as

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}; \quad S^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}. \quad (2.11)$$

Table 2.1: The different circuits for measuring in each of the three Pauli X , Y , and Z bases. Measurement outcomes are given in logical 0 and logical 1 corresponding to the $+1$ and -1 eigenstates of the bases.

MEASUREMENT BASIS	CIRCUIT	EFFECTIVE MEASUREMENT BASIS	MEASUREMENT OUTCOME	$ \psi\rangle =$
X			0	$ +\rangle$
			1	$ -\rangle$
Y			0	$ y+\rangle$
			1	$ y-\rangle$
Z			0	$ 0\rangle$
			1	$ 1\rangle$

2.2.4 Entropy

The von Neumann entropy is the analogous entropic relation to the Shannon entropy.

It is given by

$$S(\rho) = -\text{tr}(\rho \log_2 \rho). \quad (2.12)$$

ρ should be a diagonalisable matrix. If not given in the form of a diagonal matrix, ρ has to be diagonalised into a diagonal matrix before Eq. (2.12) can be computed.

It is possible to formulate the joint and conditional quantum entropies and the mutual information as well, but for the purpose of this thesis, we need not find such quantities and will refer the inquisitive reader to Ref. [44] for more information.

2.2.5 Pauli transfer matrix

The Pauli transfer matrix notation is used for gate set tomography [45], part of the procedure for error mitigation technique in Chapter 6. In the Pauli transfer matrix (PTM) notation, a quantum state ρ is written as a column vector

$$|\rho\rangle\rangle \equiv [\dots \rho_\sigma \dots]^T \quad (2.13)$$

with each ρ_σ is given by $\rho_\sigma \equiv \text{tr}(\sigma \rho)$. Here, ρ is the density matrix while $\sigma \in \{I, \sigma_x, \sigma_y, \sigma_z\}^{\otimes n}$ are the Pauli operators. n denotes the number of qubits for the system.

Operators that act on ρ transform ρ into another state. These operators are usually referred to as quantum channels and are given by $\mathcal{O}(\rho) = \sum_k E_k \rho E_k^\dagger$, where E_k are Kraus operators, $\sum_k E_k^\dagger E_k = I$. $\mathcal{O}$ are completely positive trace-preserving maps. The resulting quantum channel in Pauli transfer matrix notation $\mathcal{O}^{\text{PTM}}$ is a real square matrix with elements

$$\mathcal{O}_{\sigma,\tau}^{\text{PTM}} \equiv \frac{1}{2^n} \text{tr}[\sigma \mathcal{O}(\tau)] \quad (2.14)$$

where $\sigma, \tau \in \{I, \sigma_x, \sigma_y, \sigma_z\}^{\otimes n}$ are Pauli operators. As such, $\rho' \equiv \mathcal{O}(\rho)$ is akin to $|\rho'\rangle\rangle = |\mathcal{O}(\rho)\rangle\rangle = \mathcal{O}^{\text{PTM}}|\rho\rangle\rangle$ in Pauli transfer matrix notation.

Measurements M in the Pauli transfer matrix notation are row vectors

$$\langle\langle M | \equiv [\dots M_\sigma \dots] \quad (2.15)$$

where $M_\sigma \equiv \frac{1}{2^n} \text{tr}(\sigma M)$. σ again is the set of Pauli operators. The expectation value of measurement M is given as $\langle M \rangle = \text{tr}(M\rho)$ by Born's rule. In the Pauli transfer matrix notation, $\langle M \rangle = \langle\langle M | \rho \rangle\rangle$.

2.3 Stochastic Processes

Classical dynamical systems do not collapse under measurement like quantum systems. Learning about its behaviour is as straightforward as probing the system at every t time interval and recording its state. The collection of the outcomes can be treated as a stochastic process, or a time series, which indicates how the system evolves with time. At every time step t , the stochastic process is described with a random variable X_t whose observed values x_t are taken from a set of finite symbols $\mathcal{A}$. The number of elements in the set $\mathcal{A}$ is given by its cardinality, $|\mathcal{A}|$.

To describe L consecutive observables, we make use of the shorthand notation $x_{0:L} \equiv x_0, x_1, \dots, x_{L-1}$. The left subscript is inclusive while the right subscript is exclusive. $x_{0:L}$ may be referred to as an L -length sequence or L -length *word*.

After probing the system for an infinite amount of time, one would have an infinite stochastic process. Every time step t of the infinite process has its individual (infinite) past and future, represented with semi-infinite halves $\overleftarrow{x} \equiv x_{-\infty:t}$ and $\overrightarrow{x} \equiv x_{t:\infty}$. These stochastic processes we are interested in have a property known as stationarity. Stationarity implies that the statistical distribution of the process does not change

with time and is translationally invariant,

$$P(x_{0:\zeta}) = P(x_{\kappa:\kappa+\zeta}) \quad (2.16)$$

where κ and ζ are real and positive.

Typically, a stochastic process that is not identically and independently distributed will have some degree of correlations between the past and future. The amount of correlation can be quantified with the mutual information between the past and future. From an operational perspective, the mutual information indicates the minimum amount of information a model requires to generate statistically precise futures.

When presented with a stochastic process such as a time series, one can employ frequency counting to build up probabilities and multiply them using the following equation for a single outcome of the stochastic process $\vec{x} = x_{-\infty:\infty}$

$$\begin{aligned} I(\overleftarrow{x}; \overrightarrow{x}) &= \lim_{L \rightarrow \infty} I(x_{-L:0}; x_{0:L}) \\ &= \lim_{L \rightarrow \infty} \sum_{x_{-L:0}, x_{0:L} \in \mathcal{A}^L} P(x_{-L:0}, x_{0:L}) \log_2 \left[\frac{P(x_{-L:0}, x_{0:L})}{P(x_{-L:0})P(x_{0:L})} \right] \end{aligned} \quad (2.17)$$

Frequency counting with a finite stochastic process gives inexact probability distributions that will asymptote towards the exact distributions according to the law of large numbers [46].

CHAPTER 3

Causal models

*We know the past but cannot control it.
We control the future but cannot know it.*

- Claude E. Shannon

Predicting the futures of stochastic processes has been a longstanding task in quantitative science. These are often complex processes that exist everywhere in naturally occurring processes such as the weather [47, 48] and DNA sequences [49] as well as artificial processes like the stock market [50] and traffic [51]. Modelling such processes aids in helping one to understand the underlying dynamics of the process. With a properly established model, one would have the potential to obtain future predictions of these processes. The question that faces us is: what model would be best suited for the task?

This Chapter builds upon the philosophy of Occam's Razor which emphasises that *"It is vain to do with more what can be done with less"* [52], or in a more pertinent

scenario, *“Scientists must use the simplest means of arriving at their results and exclude everything not perceived by the senses”* [53]. Occam’s Razor may be translated into the following for the challenge of predicting the future of a stochastic process, *the task of understanding the dynamics of a stochastic process and predicting its future should be solved with as few assumptions as possible*. Occam’s Razor provides an inspiration to build a model that requires the minimal amount of information used to track the past of a process but is able to statistically reconstruct it.

We simplify our problem with the assumption that the stochastic process presented is in the form of time series data that is not affected by any external factors. For example, by observing an ideal oscillator’s displacement, the model should be able to store the displacement values in some memory and derive the oscillator’s dynamics.

Modelling time series has added value if the model carries an operational approach to understanding the time series. The model should be able to run and produce another time series that is statistically identical to its input. Herein lies the challenge of building such models. In the field of complexity science, computational mechanics allows this very act of compactly representing a time series with a model. It asks, what is the minimal amount of information one should track of the process’ past in order to predict the future faithfully?

3.1 Classical causal models

Classical causal models tackle the modelling problem from the perspective of an observer who is tasked to observe sequences and statistically predict the future. Instead of the brute force approach of remembering all possible pasts and their futures, which quickly spirals into needing an unbounded amount of memory, computational mechanics groups pasts that have the same probability of leading to the same futures in an equivalence class.

Take the fair coin toss as an example. Since all pasts have statistically identical futures, computational mechanics dictates that all pasts should be grouped together in one equivalence class.

This grouping of pasts can be mathematically captured by the equivalence relation,

$$\overleftarrow{x} \sim \overleftarrow{x}' \iff P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}) = P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}'). \quad (3.1)$$

As long as $P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}) = P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}')$, the two pasts $\overleftarrow{x}$ and $\overleftarrow{x}'$ are said to be equivalent and grouped into the same equivalence class. Parsing through the entire process then allows the process to be segregated in various equivalence classes which are also called causal states, $\{s_{\overleftarrow{x}}\} \equiv \mathcal{S}$.

Figure 3.1: Equivalence classes are formed by grouping pasts with statistically identical futures.

The stochastic process goes from $\overleftarrow{x}$ to $\overleftarrow{x}x$ to $\overleftarrow{x}xx$ and so on. The sequence $\overleftarrow{x}$ is assigned to equivalence class $s_{\overleftarrow{x}}$ which we relabel as s_j . The next sequence is assigned to equivalence class $s_{\overleftarrow{x}x}$ which we relabel as s_k . This allows one to make the connection that there are transitions between equivalence classes that occur probabilistically with some output x and probability p . We use a deterministic update function $\lambda(x, j)$ to denote the label of the updated equivalence class when s_j outputs x . The deterministic function enables *unifilarity*. Unifilarity is an important property that allows the process to be represented as a deterministic edge-emitting hidden Markov model known as the ϵ -machine (see Fig. 3.2 for an example of an ϵ -machine).

For models which are not unifilar, there exists methods to *unifilarise* them through mixed-state decomposition and reconstruction [54, 55]

Figure 3.2: An example of an ϵ -machine. The nodes represent causal states and directed edges indicate the transitions between causal states. Each transition occurs with some probability p and is accompanied with an output x . The statistical complexity of the ϵ -machine is defined as the Shannon entropy over the stationary distribution of the causal states.

After establishing the set of equivalence classes and the transitions between equivalence classes, one would only have to track which causal state the process is in instead of tracking every past $\overleftarrow{x}$ of the process. The subsequent step is to look at the amount of information that the ϵ -machine contains. The statistical complexity is defined as the Shannon entropy of the stationary distribution of causal states. Solving the eigenvalue equation $T\vec{P} = \vec{P}$ with T being the stochastic transition matrix containing all state-to-state transition probabilities and $\vec{P}$ is the vector containing the probabilities of each causal state, the statistical complexity is

$$C_\mu \equiv H[P(s_j)] = - \sum_j P(s_j) \log_2 P(s_j). \quad (3.2)$$

The statistical complexity carries various interpretations.

- (1) The statistical complexity quantifies the amount of resources or memory that the ϵ -machine needs to store about the process in order to statistically predict faithful futures.
- (2) The statistical complexity is the amount of information one gains upon learning a process' current causal state. This is closely related to the amount of

information that two observers with their individual (identical) ε -machines need to communicate with each other in order to synchronise their ε -machines [21, 22].

- (3) The statistical complexity indicates the amount of structure that a process contains that can be captured in the causal states [11, 13, 56].

To this end, there are three known methods of reconstructing an ε -machine of a stochastic process:

- (i) Sub-tree reconstruction algorithm [11].
- (ii) Causal state splitting reconstruction algorithm [14].
- (iii) Bayesian structural inference [57].

Since the sub-tree reconstruction algorithm is used for the results in Chapter 7, its methodology will be reviewed in Section 3.1.1.

3.1.1 How to build an ε -machine

In this segment, we explicitly describe the five main steps to build an ε -machine using the sub-tree reconstruction algorithm.

1. Constructing a tree structure. A moving window of size $2L$ is first chosen to parse through the entire stochastic process. At the beginning of the stochastic process, the first moving window gives rise to a blank node that signifies the start of the process. From this blank node, every $2L$ successive nodes are created with a directed link for every $x_{0:2L}$. For any $x_{0:2L}$ sequence that can be traced with existing directed links and nodes, no new links and nodes are added. New links and nodes are added only when no existing path can be traced.

For example, suppose $x_{0:6} = 111101$ gives rise to six new nodes that branch from the blank node. If $x_{1:7} = 111010$, then no new nodes are drawn for $x_{1:6} = 11101$ but

Figure 3.3: The sub-tree reconstruction algorithm, here illustrated for $L = 3$.

a new node is drawn for $x_6 = 0$. Each different element of the $|\mathcal{A}|^{2L}$ total number of combinations of $x_{0:2L}$ gives rise to an individual set of directed links and nodes, implying that there are a maximum of $|\mathcal{A}|^{2L}$ branches originating from the blank node.

2. Assigning probabilities. The initial blank node emits outputs $x \in \mathcal{A}$. The probability that the initial blank node emits output x can be computed by taking the ratio of the number of branches that x emits out of $\mathcal{A}$. For subsequent nodes, the probability of outputting x can be computed by taking the ratio of the number of branches the particular node emits x out of the total number of branches within that particular node. This allows each link to be labelled with an output x with some probability p .

For example, if we let $x \in \mathcal{A} = \{0, 1\}$, then the probabilities that the node outputs

x is given by

$$\begin{aligned} P(x = 0) &= \frac{\text{Number of nodes reached from } x = 0}{\text{Total number of nodes reached from } x = 0 \text{ and } x = 1}; \\ P(x = 1) &= \frac{\text{Number of nodes reached from } x = 1}{\text{Total number of nodes reached from } x = 0 \text{ and } x = 1}. \end{aligned} \quad (3.3)$$

3. Comparing sub-trees. Starting from the initial blank node, the tree structure reached by L outputs is compared against all other nodes. All nodes with identical $x|p$ and branch structure of L sizes are then given the same label. In reality, stochastic processes are of finite length. L has to be finite as well. This implies that there will be a finite number of branches, which in turn results in the probabilities that a node emits x to be inexact. Therefore, a χ^2 test with standard tolerance 0.05 is used to account for these statistical artefacts. The χ^2 test aids in labelling nodes that have similar-enough branch structures, essentially enforcing the causal equivalence relation. One has freedom to adjust the tolerance value.

Given a stochastic process that is illustrated in Fig. 3.3, should the tree structure be identical and that p be close enough to q where the χ^2 test determines the starting node and the following node (on the left) under it are similar enough, they will be given the same label s_A .

4. Constructing the ε -machine. Each individually labelled node with their single output and transition probability is analysed to form a model with nodes and transitions. This model represents an edge-emitting hidden Markov model of the process and it is the inferred ε -machine of the stochastic process.

5. Computing the statistical complexity. Node-to-node transition probabilities can be recorded into a transition matrix T . Solving the eigenvalue equation $T\vec{P} = \vec{P}$ gives the vector $\vec{P}$ containing the steady-state probabilities of the causal states. The

statistical complexity can then be calculated by

$$C_\mu \equiv H[P(s_j)] = - \sum_j P(s_j) \log_2 P(s_j), \quad (3.4)$$

where $P(s_j)$ make up the elements within $\vec{P}$ and $\sum_j P(s_j) = 1$.

3.1.2 C_μ vs mutual information

We remarked that the mutual information of a stochastic process quantifies the amount of correlations between the past and future of a stochastic process. This term has been referred to as the excess entropy $E \equiv I(\overleftarrow{X}; \overrightarrow{X})$. How does that relate to C_μ ?

Since causal models are effectively models of stochastic processes, the best models should require a minimum of E bits to statistically recreate the process. However, the inevitable inequality occurs $E < C_\mu$ with the difference $|E - C_\mu|$ being termed as modelling overhead [58]. The modelling overhead is attributed to the overlaps of the distribution of futures that the causal states generate.

Summary for classical causal models

Classical causal models, also known as ε -machines, are a useful tool to understand the underlying dynamics of a stochastic process. The states within a classical causal model are known as causal states and can be found by grouping pasts with statistically identical futures together, i.e. the equivalence relation:

$$\overleftarrow{x} \sim \overleftarrow{x}' \iff P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}) = P(\overrightarrow{X} = \overrightarrow{x} | \overleftarrow{x}'). \quad (3.5)$$

The statistical complexity is defined as the Shannon entropy of the stationary distribution of causal states and is given by

$$C_\mu \equiv H[P(s_j)] = - \sum_j P(s_j) \log_2 P(s_j) \quad (3.6)$$

Such classical causal models can be constructed with various classical inference algorithms such as the sub-tree reconstruction algorithm, the causal state splitting reconstruction algorithm, and the Bayesian statistical inference algorithm.

Typically, classical causal models incur a modelling overhead that results in the inequality $E < C_\mu$, where E is the excess entropy.

3.2 Quantum causal models

3.2.1 Version 1: Kraus operations (2012)

The application of quantum mechanics to existing classical causal models allows a decrease memory requirement while retaining predictive capabilities. The modelling overhead given by $|E - C_\mu|$ can be reduced. The first version of quantum causal models involves straightforwardly mapping each causal state in the ε -machine to a quantum state [19, 20] to build an analogue quantum causal model.

The probability of transition is given by $T_{j \rightarrow k}^{(x)} \equiv P(s_k | x, s_j)$, now written under square-roots as probability amplitudes. This is the conservation of probability, $\sum_j \left| \sqrt{T_{j \rightarrow k}^{(x)}} \right|^2 = 1$. In the quantum regime the quantum states are defined as

$$|s_j\rangle = \sum_k \sum_{x \in \mathcal{A}} \sqrt{T_{j \rightarrow k}^{(x)}} |x\rangle |k\rangle. \quad (3.7)$$

$|k\rangle$ represents the label of the next state $|s_k\rangle$. Transforming $|k\rangle$ to $|s_k\rangle$ allows one to extract $|x\rangle$, the output from the transition.

Each operation to extract $|x\rangle$ involves appending an ancillary qubit $|0\rangle$ to Eq. 3.7 to obtain the following intermediary state

$$|s'_j\rangle = \sum_k \sum_{x \in \mathcal{A}} \sqrt{T_{j \rightarrow k}^{(x)}} |x\rangle |0\rangle |k\rangle, \quad (3.8)$$

then apply a Kraus operation $K = \sum_k |s_k\rangle \langle 0| \langle k|$ that selects the state $|s_k\rangle$ from joint state $|0\rangle |k\rangle$. Measuring $|x\rangle$ in the computational basis would yield the output x . This leaves $|s_k\rangle$ to be sent back to the system for repeated application.

The quantum statistical memory of such a process was defined using the von

Neumann entropy of $\{|s_j\rangle\}$. It is given by

$$C_q \equiv S[\rho] = -\text{tr}(\rho \log_2 \rho) \quad (3.9)$$

where $\rho = \sum_j P(s_j) |s_j\rangle \langle s_j|$ is the density matrix of the quantum states. Here, $P(s_j) \equiv P(|s_j\rangle)$ due to the direct one-to-one mapping from a causal state to a quantum state. Because of the non-orthogonality of quantum states, i.e. $0 < |\langle s_j | s_k \rangle| < 1$, $C_q < C_\mu$. $C_q = C_\mu$ only holds when all quantum states $\{|s_j\rangle\}$ are orthogonal. With the modelling overhead reduced, the quantum statistical memory of quantum causal models fulfil the following condition $E \leq C_q \leq C_\mu$. This memory advantage of quantum causal models over their classical counterparts can be illustrated with an example known as the perturbed coin process.

Here and throughout this thesis, we avoid the term ‘quantum statistical complexity’ as it is solely reserved for the best quantum causal models where C_q is absolutely minimum. We will see in the subsequent versions of quantum causal models that C_q *can* be reduced further. Therefore, we will use the term ‘quantum statistical memory’ to denote the memory requirements of quantum causal models here and throughout this thesis.

Example: Perturbed coin process Figure 3.4: The ϵ -machine for the perturbed coin process.

The perturbed coin process features two causal states with inter-state transitions given by $x|$ probability. It is modelled after a Markovian process; the causal state upon learning output $x = 0$ ($x = 1$) leads to a state s_A (s_B). At $0 \leq p \leq 1$, $p \neq 0.5$, the ϵ -machine for the perturbed coin process has stochastic transition matrix

$$T = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}. \quad (3.10)$$

We solve the eigenvalue equation to obtain the eigenvector that corresponds to the stationary distribution of the causal states s_A and s_B .

$$\begin{aligned} T\vec{P} &= \vec{P} \\ \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix} \begin{pmatrix} P(s_A) \\ P(s_B) \end{pmatrix} &= \begin{pmatrix} P(s_A) \\ P(s_B) \end{pmatrix} \\ \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix} &= \begin{pmatrix} P(s_A) \\ P(s_B) \end{pmatrix} \end{aligned} \quad (3.11)$$

$P(s_A)$ and $P(s_B)$ are equiprobable. Thus, the ε -machine's statistical complexity is

$$\begin{aligned} C_\mu &= -P(s_A) \log_2 P(s_A) - P(s_B) \log_2 P(s_B) \\ &= -(0.5) \log_2(0.5) - (0.5) \log_2(0.5) \\ &= 1. \end{aligned} \tag{3.12}$$

At $p = 0.5$, since both causal states have identical outputs and transition probabilities, it is no longer the minimal model according to the equivalence relation (Eq. (3.5)). The minimal model would require s_A and s_B to be merged into a single causal state with $p(x = 0) = p(x = 1) = 0.5$. The resulting C_μ of an ε -machine with a single causal state is $C_\mu = 0$. Therefore, the ε -machine has $C_\mu = 1$ at $0 \leq p \leq 1$, $p \neq 0.5$, and $C_\mu = 0$ at $p = 0.5$.

Applying the direct one-to-one mapping of causal states to quantum states one would obtain the following quantum states,

$$\begin{aligned} |s_A\rangle &= \sqrt{1-p} |x=0\rangle |k=1\rangle + \sqrt{p} |x=1\rangle |k=2\rangle \\ |s_B\rangle &= \sqrt{p} |x=0\rangle |k=1\rangle + \sqrt{1-p} |x=1\rangle |k=2\rangle. \end{aligned} \tag{3.13}$$

Because output $x = 0$ always leads to state s_A and output $x = 1$ always leads to state s_B , we apply the short hand notation $|x\rangle |k\rangle = |xk\rangle = |\bar{x}\rangle$. This results in

$$\begin{aligned} |s_A\rangle &= \sqrt{1-p} |x=0, k=1\rangle + \sqrt{p} |x=1, k=2\rangle \\ &= \sqrt{1-p} |\bar{0}\rangle + \sqrt{p} |\bar{1}\rangle \\ |s_B\rangle &= \sqrt{p} |x=0, k=1\rangle + \sqrt{1-p} |x=1, k=2\rangle \\ &= \sqrt{p} |\bar{0}\rangle + \sqrt{1-p} |\bar{1}\rangle. \end{aligned} \tag{3.14}$$

The resultant density matrix is

$$\begin{aligned}
 \rho &= P(s_A) |s_A\rangle \langle s_A| + P(s_B) |s_B\rangle \langle s_B| \\
 &= P(s_A) [(1-p) |\bar{0}\rangle \langle \bar{0}| + \sqrt{p(1-p)} |\bar{0}\rangle \langle \bar{1}| + \sqrt{p(1-p)} |\bar{1}\rangle \langle \bar{0}| + p |\bar{1}\rangle \langle \bar{1}|] + \quad (3.15) \\
 &\quad P(s_B) [(p) |\bar{0}\rangle \langle \bar{0}| + \sqrt{p(1-p)} |\bar{0}\rangle \langle \bar{1}| + \sqrt{p(1-p)} |\bar{1}\rangle \langle \bar{0}| + (1-p) |\bar{1}\rangle \langle \bar{1}|].
 \end{aligned}$$

For simplicity, we let $|\bar{0}\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|\bar{1}\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, we now have

$$\begin{aligned}
 \rho &= P(s_A) \left[(1-p) \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \sqrt{p(1-p)} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \sqrt{p(1-p)} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} + p \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \right] + \\
 &\quad P(s_B) \left[(p) \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \sqrt{p(1-p)} \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \sqrt{p(1-p)} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} + (1-p) \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \right] \\
 &= P(s_A) \begin{pmatrix} 1-p & \sqrt{p(1-p)} \\ \sqrt{p(1-p)} & p \end{pmatrix} + P(s_B) \begin{pmatrix} p & \sqrt{p(1-p)} \\ \sqrt{p(1-p)} & (1-p) \end{pmatrix}. \quad (3.16)
 \end{aligned}$$

Substituting $P(s_A) = P(s_B) = 0.5$ and diagonalising ρ , we have

$$\rho_{\text{diag}} = \begin{pmatrix} \frac{1}{2} + \sqrt{p(1-p)} & 0 \\ 0 & \frac{1}{2} - \sqrt{p(1-p)} \end{pmatrix}. \quad (3.17)$$

The corresponding analytical solution shows that the quantum statistical memory is continuous with p ,

$$\begin{aligned}
 C_q &= -\text{tr}(\rho_{\text{diag}} \log_2 \rho_{\text{diag}}) \\
 &= -\left(\frac{1}{2} + \sqrt{p(1-p)}\right) \log_2 \left(\frac{1}{2} + \sqrt{p(1-p)}\right) \\
 &\quad -\left(\frac{1}{2} - \sqrt{p(1-p)}\right) \log_2 \left(\frac{1}{2} - \sqrt{p(1-p)}\right). \quad (3.18)
 \end{aligned}$$

Figure 3.5: The blue (red) line represents how C_q (C_μ) scales with p . Note that at $p = 0.5$, $C_\mu = 0$.

Both C_μ and C_q are plotted in Fig. 3.5 in the range $0 \leq p \leq 1$ for a visual comparison.

For $0 < p < 1$ and $p \neq 0.5$, the inequality $C_q < C_\mu$ holds. $\vec{P}$ is independent of p as seen in Eq. (3.11), giving $C_\mu = 1$. At $p = 0.5$, the causal states s_A and s_B are indistinguishable due to them outputting the same futures with identical probabilities. As such, they can be merged into one single causal state. The statistical complexity of a single causal state is $C_\mu = 0$.

Plotting C_q and C_μ on a graph (Fig. 3.5) with $0 \leq p \leq 1$, we observe that the biggest quantum advantage occurs when $p = 0.5 \pm \delta$ for some $\delta \ll 1$ where $C_\mu = 1$ and $C_q \approx 0$.

We omit plots of the excess entropy as this example focuses on the memory advantage in which quantum causal models have over the ε -machine. Nonetheless, the relationship between excess entropy and statistical complexity/memory can be found in Fig. 2 of Ref. [19].

3.2.2 Version 2: Cryptic order (2016)

A separate study found that by increasing each states' futures from one time step in Eq. (3.7) to multiple time steps up to the cryptic order, the non-orthogonality between quantum states can be increased further, allowing for a greater reduction in the modelling overhead [21]. The cryptic order describes the minimum length of past observations required to be certain of the present causal state, given that one knows the entire future, i.e., the minimum k such that $H(s_k|x_{0:\infty}) = 0$. It was shown that if quantum states encode future L -length words that have $1 < L \leq k$, non-orthogonality between quantum states is increased and greater compression of memory will be achieved. Hence, the cryptic order also represents the minimum future L -length for the greatest non-orthogonality between quantum states. In this formalism, the string of consecutive future outputs $|x_0\rangle|x_1\rangle|x_2\rangle, \dots, |x_{L-1}\rangle$ are encoded as L -length words, $x_{0:L} \equiv w^L$. The maximum compression in memory is achieved when $L = k$, with k being the cryptic order of the process.

These quantum states are given as

$$|\eta_j(L)\rangle = \sum_{w^L \in \mathcal{A}^L} \sum_{S_k \in \mathcal{S}} \sqrt{P(w^L, \sigma_k | \sigma_j)} |s_k\rangle |w^L\rangle \quad (3.19)$$

where $|w^L\rangle \equiv |x_0\rangle|x_1\rangle|x_2\rangle \dots |x_{L-1}\rangle$. Likewise, the corresponding statistical memory is also given by the von Neumann entropy of $\{|\eta_j(L)\rangle\}$ as

$$C_q(L) = -\text{tr}(\rho(L) \log_2 \rho(L)) \quad (3.20)$$

where $\rho(L) = \sum_j P(\eta_j) |\eta_j(L)\rangle \langle \eta_j(L)|$. $P(\eta_j)$ is equivalent to the probability of classical causal state $P(s_j)$ as the one-to-one mapping of classical causal states to quantum states is still used.

Increasing the future word length increases the amount of overlaps between

the quantum states provided the cryptic order $k > 1$ and Eq. (3.19) incorporates $(1 < L \leq k)$ -length words. Representing the quantum states with future L -length words as $|\eta_j(L)\rangle$ (from Eq. 3.19) and a single future time step as $|s_j\rangle$ (from Eq. 3.7), we have

$$\langle \eta_j(L) | \eta_k(L) \rangle \leq \langle s_j | s_k \rangle \leq 1. \quad (3.21)$$

The quantum statistical memory computed from Eq. (3.9) is based strictly on $L = 1$ futures and we denote it as $C_q(L = 1)$. Comparing all memory requirements thus far, we have

$$C_q(1 < L \leq k) \leq C_q(L = 1) \leq C_\mu. \quad (3.22)$$

We will use the example of the 4-3 Golden Mean Process to illustrate that increasing the future word lengths results more memory compression [21].

Example: 4-3 Golden Mean Process

Figure 3.6: The ϵ -machine for the 4-3 Golden Mean Process.

In this example, we determine the quantum states for various future lengths from one time step to beyond the cryptic order. We will show that $C_q(L)$ reaches a minimum and plateaus when L is equal to the cryptic order k . C_μ will not change as it only depends on the stationary distribution of causal states and does not factor in the cryptic order. To find C_μ , we first obtain the transition matrix then solve the

eigenvalue equation $T\vec{P} = \vec{P}$. The transition matrix for the 4-3 Golden Mean Process is

$$T = \begin{pmatrix} p & 0 & 0 & 0 & 0 & 0 & 1 \\ 1-p & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \quad (3.23)$$

The corresponding vector $\vec{P}$ containing the stationary probabilities of the causal states is found to be

$$\vec{P} = \begin{pmatrix} 1/(7-6p) \\ (1-p)/(7-6p) \\ (1-p)/(7-6p) \\ (1-p)/(7-6p) \\ (1-p)/(7-6p) \\ (1-p)/(7-6p) \\ (1-p)/(7-6p) \end{pmatrix}. \quad (3.24)$$

The classical statistical complexity can be straightforwardly calculated as

$$\begin{aligned} C_\mu &= - \sum_j P(s_j) \log_2 P(s_j) \\ &= - \left(\frac{1}{7-6p} \right) \log_2 \left(\frac{1}{7-6p} \right) - 6 \left(\frac{1-p}{7-6p} \right) \log_2 \left(\frac{1-p}{7-6p} \right). \end{aligned} \quad (3.25)$$

We now determine the quantum states directly from the 4-3 Golden Mean Process.

At $L = 1$, the seven quantum states are

$$\begin{aligned}
 |\eta_1(L = 1)\rangle &= \sqrt{p}|s_A\rangle|0\rangle + \sqrt{1-p}|s_B\rangle|1\rangle \\
 |\eta_2(L = 1)\rangle &= |s_3\rangle|1\rangle \\
 |\eta_3(L = 1)\rangle &= |s_4\rangle|1\rangle \\
 |\eta_4(L = 1)\rangle &= |s_5\rangle|1\rangle \\
 |\eta_5(L = 1)\rangle &= |s_6\rangle|0\rangle \\
 |\eta_6(L = 1)\rangle &= |s_7\rangle|0\rangle \\
 |\eta_7(L = 1)\rangle &= |s_A\rangle|0\rangle
 \end{aligned} \tag{3.26}$$

At $L = 2$,

$$\begin{aligned}
 |\eta_1(L = 2)\rangle &= \sqrt{p^2}|s_A\rangle|00\rangle + \sqrt{p(1-p)}|s_B\rangle|01\rangle + |1-p|s_3\rangle|11\rangle \\
 |\eta_2(L = 2)\rangle &= |s_4\rangle|11\rangle \\
 |\eta_3(L = 2)\rangle &= |s_5\rangle|11\rangle \\
 |\eta_4(L = 2)\rangle &= |s_6\rangle|10\rangle \\
 |\eta_5(L = 2)\rangle &= |s_7\rangle|00\rangle \\
 |\eta_6(L = 2)\rangle &= |s_A\rangle|00\rangle \\
 |\eta_7(L = 2)\rangle &= \sqrt{p}|s_A\rangle|00\rangle + \sqrt{1-p}|s_B\rangle|01\rangle
 \end{aligned} \tag{3.27}$$

At $L = 3$,

$$\begin{aligned}
|\eta_1(L = 3)\rangle &= \sqrt{p^3}|s_A\rangle|000\rangle + \sqrt{p^2(1-p)}|s_B\rangle|001\rangle + |p(1-p)\rangle|s_3\rangle|011\rangle \\
&\quad + \sqrt{1-p}|s_4\rangle|111\rangle \\
|\eta_2(L = 3)\rangle &= |s_5\rangle|111\rangle \\
|\eta_3(L = 3)\rangle &= |s_6\rangle|110\rangle \\
|\eta_4(L = 3)\rangle &= |s_7\rangle|100\rangle \\
|\eta_5(L = 3)\rangle &= |s_A\rangle|000\rangle \\
|\eta_6(L = 3)\rangle &= \sqrt{p}|s_A\rangle|000\rangle + \sqrt{1-p}|s_B\rangle|001\rangle \\
|\eta_7(L = 3)\rangle &= \sqrt{p^2}|s_A\rangle|000\rangle + \sqrt{p(1-p)}|s_B\rangle|001\rangle + |1-p\rangle|s_3\rangle|011\rangle
\end{aligned} \tag{3.28}$$

At $L = 4$,

$$\begin{aligned}
|\eta_1(L = 4)\rangle &= \sqrt{p^4}|s_A\rangle|0000\rangle + \sqrt{p^3(1-p)}|s_B\rangle|0001\rangle + |p^2(1-p)\rangle|s_3\rangle|0011\rangle \\
&\quad + \sqrt{p(1-p)}|s_4\rangle|0111\rangle + \sqrt{1-p}|s_5\rangle|1111\rangle \\
|\eta_2(L = 4)\rangle &= |s_5\rangle|1110\rangle \\
|\eta_3(L = 4)\rangle &= |s_6\rangle|1100\rangle \\
|\eta_4(L = 4)\rangle &= |s_7\rangle|1000\rangle \\
|\eta_5(L = 4)\rangle &= \sqrt{p}|s_A\rangle|0000\rangle + \sqrt{1-p}|s_B\rangle|0001\rangle \\
|\eta_6(L = 4)\rangle &= \sqrt{p^2}|s_A\rangle|0000\rangle + \sqrt{p(1-p)}|s_B\rangle|0001\rangle + |1-p\rangle|s_3\rangle|0011\rangle \\
|\eta_7(L = 4)\rangle &= \sqrt{p^3}|s_A\rangle|0000\rangle + \sqrt{p^2(1-p)}|s_B\rangle|0001\rangle + |p(1-p)\rangle|s_3\rangle|0011\rangle \\
&\quad + \sqrt{1-p}|s_4\rangle|0111\rangle
\end{aligned} \tag{3.29}$$

At $L = 5$,

$$\begin{aligned}
|\eta_1(L = 5)\rangle &= \sqrt{p^5}|s_A\rangle|00000\rangle + \sqrt{p^4(1-p)}|s_B\rangle|00001\rangle + |p^3(1-p)\rangle|s_3\rangle|00011\rangle \\
&\quad + \sqrt{p^2(1-p)}|s_4\rangle|00111\rangle + \sqrt{p(1-p)}|s_5\rangle|01111\rangle \\
&\quad + \sqrt{1-p}|s_6\rangle|11110\rangle \\
|\eta_2(L = 5)\rangle &= |s_5\rangle|11100\rangle \\
|\eta_3(L = 5)\rangle &= |s_6\rangle|11000\rangle \\
|\eta_4(L = 5)\rangle &= \sqrt{p}|s_A\rangle|10000\rangle + \sqrt{1-p}|s_B\rangle|10001\rangle \\
|\eta_5(L = 5)\rangle &= \sqrt{p^2}|s_A\rangle|00000\rangle + \sqrt{p(1-p)}|s_B\rangle|00001\rangle + |1-p\rangle|s_3\rangle|00011\rangle \\
|\eta_6(L = 5)\rangle &= \sqrt{p^3}|s_A\rangle|00000\rangle + \sqrt{p^2(1-p)}|s_B\rangle|00001\rangle + |p(1-p)\rangle|s_3\rangle|00011\rangle \\
&\quad + \sqrt{1-p}|s_4\rangle|00111\rangle \\
|\eta_7(L = 5)\rangle &= \sqrt{p^4}|s_A\rangle|00000\rangle + \sqrt{p^3(1-p)}|s_B\rangle|00001\rangle + |p^2(1-p)\rangle|s_3\rangle|00011\rangle \\
&\quad + \sqrt{p(1-p)}|s_4\rangle|00111\rangle + \sqrt{1-p}|s_5\rangle|01111\rangle
\end{aligned} \tag{3.30}$$

Since the cryptic order $k = 3$, stopping at $L = 5$ would provide a sufficient runway to illustrate how $C_q(L)$ plateaus when L reaches the cryptic order.

Now, one can easily compute the density matrix $\rho(L)$ for each set of quantum states corresponding to each future L -length with the quantum states using $\vec{P}$ as the stationary distribution of causal states. The resulting $C_q(L)$ can be worked out easily. The results can be plotted in Fig. 3.7. Irrespective of the parameter p ($0 < p < 1$), observe that $C_q(L)$ decreases with increasing L and plateaus when L reaches the cryptic order at $k = 3$.

For completeness, to see how C_μ stacks against the various $C_q(L)$ at different future L -lengths, we plot Fig. 3.8 with $p = 0.2$. Note that C_μ shows no change with L as C_μ is solely based on each causal state's probability of occurrence.

Figure 3.7: The 4-3 Golden Mean Process. Analytical results for $C_q(L)$ with various values of p are plotted up to $L = 5$. The cryptic order is $k = 3$. Observe that $C_q(L)$ plateaus when $L \geq k$.

Figure 3.8: The red line represents $C_q(L)$ while the blue line represents C_μ , both computed with the 4-3 Golden Mean Process set to $p = 0.2$.

3.2.3 Version 3: Unitary quantum causal models (2018 onwards)

Figure 3.9: The unitary operator acts on the quantum memory state $|\sigma_j\rangle$ with a blank ancillary state to produce x when measured in the appropriate basis.

With quantum computing on the horizon, there is great interest in using quantum computers to solve tasks. Unitary quantum causal models were developed to replace existing quantum causal models. In this framework, the quantum states are $|\sigma_j\rangle$ and satisfy the following unitary evolution [23]

$$U|\sigma_j\rangle|0\rangle = \sum_{x \in \mathcal{A}} \sqrt{P(x|j)} |\sigma_{\lambda(x,j)}\rangle |x\rangle. \quad (3.31)$$

The system's state $|\sigma_j\rangle$ evolves under unitary operation with a blank ancillary state $|0\rangle$ to the next memory state $|\sigma_{\lambda(x,j)}\rangle$ while outputting x when the ancillary state is measured in the appropriate basis, see Fig. 3.9. The quantum causal model stores the dynamics of the stochastic process in the set of quantum states $\{|\sigma_j\rangle\}$ that forms the top register of the quantum circuit in Fig. 3.9. Thus, we will refer to $\{|\sigma_j\rangle\}$ as *quantum memory states* from hereon. The procedure for constructing the appropriate unitary operator was established in Ref. [23].

Furthermore, it was discovered that the addition of complex phases to $|\sigma_j\rangle$ gives rise to a further compression of the memory, the furthest that has ever been achieved

thus far [24]. These phase-dependent quantum memory states are

$$U|\sigma_j\rangle|0\rangle = \sum_{x \in \mathcal{A}} \sqrt{P(x|j)} e^{i\phi_{xj}} |\sigma_{\lambda(x,j)}\rangle |x\rangle. \quad (3.32)$$

Phase-dependent quantum states are given by $|\sigma_j^\phi\rangle$ with ϕ being the complex phase. These quantum states satisfy the following unitary evolution as well. As before, the quantum statistical memory is given by

$$C_q = -\text{tr}(\rho^{(\phi)} \log_2 \rho^{(\phi)}) \quad (3.33)$$

where $\rho^{(\phi)} = \sum_j P(s_j) |\sigma_j^{(\phi)}\rangle \langle \sigma_j^{(\phi)}|$. By carefully engineering the complex phases, one can achieve the maximally known compression of the memory. However, there is no known systematic method to obtain the exact complex phases as yet. In this thesis, all quantum causal models do not have complex phase, i.e. we set $\phi = 0$.

Summary for quantum causal models

Quantum causal models have seen improvements over the years. The internal quantum states of quantum causal models are obtained by a suitable one-to-one mapping of causal states (of the ε -machine) to non-orthogonal quantum states, $s_j \mapsto |\sigma_j\rangle$.

Three versions of quantum causal models exist, with each version being more efficient than its predecessor.

- Version 1: Kraus operations

$$|s_j\rangle = \sum_k \sum_{x \in \mathcal{A}} \sqrt{T_{j \rightarrow k}^{(x)}} |x\rangle |k\rangle. \quad (3.34)$$

- Version 2: Cryptic order

$$|\eta_j(L)\rangle = \sum_{w^L \in \mathcal{A}^L} \sum_{s_k \in \mathcal{S}} \sqrt{P(w^L, \sigma_k | \sigma_j)} |s_k\rangle |w^L\rangle. \quad (3.35)$$

- Version 3: Phase-dependent unitary quantum causal models

$$U|\sigma_j\rangle|0\rangle = \sum_{x \in \mathcal{A}} \sqrt{P(x|j)} e^{i\phi_{xj}} |\sigma_{\lambda(x,j)}\rangle |x\rangle. \quad (3.36)$$

The quantum model which has the least memory requirement at the time of writing are phase-dependent unitary quantum causal models. For all models, the corresponding quantum statistical memory is given by the von Neumann entropy for each of the versions' density matrices which is given by, in general, ρ .

$$C_q = -\text{tr}(\rho \log_2 \rho) \quad (3.37)$$

where the density matrix is

$$\begin{aligned} \rho &= \sum_j P(s_j) |s_j\rangle \langle s_j| \\ \rho(L) &= \sum_j P(\eta_j) |\eta_j(L)\rangle \langle \eta_j(L)| \\ \rho^{(\phi)} &= \sum_j P(s_j) |\sigma_j^{(\phi)}\rangle \langle \sigma_j^{(\phi)}| \end{aligned} \quad (3.38)$$

respectively for each version of quantum model.

CHAPTER 4

Quantum Inference Protocol

The opposite of a profound truth may well be another profound truth.

– Niels Bohr

4.1 Motivation

In Chapter [3](#), we established that the one-to-one encoding of causal states into non-orthogonal phaseless quantum states enables quantum causal models to have a memory advantage over the ϵ -machine. The addition of complex phases potentially further reduces the memory cost if the optimal complex phases are used. Are we reinventing the wheel by introducing another method of building quantum causal models when there exists a proper procedure to do so?

The motivation behind the Quantum Inference Protocol was to abolish the dependence of quantum causal models on ϵ -machines by constructing unitary quantum

Figure 4.1: The Quantum Inference Protocol analyses stochastic processes directly to build quantum causal models without any reliance on classical inference algorithms. It is expected to have similar memory requirements as the exact quantum causal models, $\tilde{C}_q \approx C_q$. Its relationship to prior quantum causal models is illustrated here.

causal models directly from the stochastic process (we may also refer to the stochastic process as *data*). Doing this sets the stage for unitary quantum causal models to be used in real-world scenarios. The major drawback in the construction of a ε -machines lies with the need for a δ -tolerance parameter that compares conditional futures to merge pasts into causal states. The need for the δ -tolerance arises when finite data causes probabilities to be inexact. Should a δ -tolerance be too lenient, pasts which are almost equivalent (according to Eq. (3.5)) but clearly different may be merged together, thereby underestimating the number of underlying causal states of the stochastic process. On the other spectrum, a δ -tolerance that is too strict will separate two supposedly equivalent pasts – two pasts with supposedly the same conditional futures that differ slightly due to small fluctuations in the inferred probabilities. Too strict a δ -tolerance overestimates the number of causal states of the stochastic process. In both scenarios, the resultant ε -machine will not be accurate. The analogous unitary quantum causal models will in turn be affected due

to the one-to-one mapping of causal states to quantum states.

The Quantum Inference Protocol was developed to negate the requirement of having the ε -machine in order to obtain the quantum causal models. In addition, it does so without a need for any δ -tolerance to merge equivalent pasts. The Quantum Inference Protocol should be able to infer the stochastic process directly to build a set of valid phaseless quantum memory states that can be used to produce statistically faithful futures. The Quantum Inference Protocol should also be able to estimate the amount of memory required for a phaseless unitary quantum model without having to construct the quantum memory states.

4.2 The set-up

We define the following quantum memory states $|\varsigma_{x_{-L:0}}\rangle$ that are built by the inference of L -length pasts $x_{-L:0}$ which satisfy the following equation under unitary evolution:

$$U|\varsigma_{x_{-L:0}}\rangle|0\rangle = \sum_{x_0 \in \mathcal{A}} \sqrt{\tilde{P}(x_0|x_{-L:0})} |\varsigma_{x_{-L+1:1}}\rangle |x_0\rangle. \quad (4.1)$$

The term under the square root is the inferred conditional probability that $x_{-L:0}$ leads to x_0 . This assumes that any L -length sequence of past leads to a causal state. We shall explore the validity of this assumption in the subsequent sections. Since a consecutive L -length sequence leads to a causal state, the probabilities of causal states can thus be computed as $\tilde{P}(x_{-L:0})$. The density matrix can then be defined as

$$\tilde{\rho}(L) = \sum_{x_{-L:0}} \tilde{P}(x_{-L:0}) |\varsigma_{x_{-L:0}}\rangle \langle \varsigma_{x_{-L:0}}|, \quad (4.2)$$

after which the inferred quantum statistical memory is obtained as

$$\tilde{C}_q(L) = -\text{tr}(\tilde{\rho}(L) \log_2 \tilde{\rho}(L)). \quad (4.3)$$

Figure 4.2: Repeated applications of the unitary operator on the memory register, now given with $|s_{x-L:0}\rangle$, with blank ancillary qubits $|0\rangle$ allows the statistical replication of the process.

Tildes are used to denote inferred quantities obtained through frequency counting.

For a model-independent method of estimating $\tilde{C}_q(L)$, one may replace the density matrix with the Gram matrix G that has elements

$$G_{x_{-L:0}x'_{-L:0}} = \sum_{x_{0:L}} \sqrt{\tilde{P}(x_{-L:0}x_{0:L})\tilde{P}(x'_{-L:0}x_{0:L})}. \quad (4.4)$$

Substituting the density matrix with the Gram matrix is possible because both have the same spectrum.

The code for the Quantum Inference Protocol is primarily written in MATLAB and has since been translated to Python. Both versions can be found at: <https://github.com/matthew0021/quantum-inference-protocol>.

In the next two sections of this Chapter, we show that the quantum memory states defined in Eq. (4.1) fulfils the following two criteria:

- (1) Prove that self merging of quantum memory states occurs at and above the Markov order (Sec. 4.3).
- (2) We then show the robustness of the quantum statistical memory. We make use of the Gram matrix to accomplish this as the Gram matrix has the same spectrum as the density matrix (Sec. 4.4).

With (1) and (2), the validity of the Quantum Inference Protocol can be established (Sec. 4.5).

4.3 Self-merging at and above the Markov order

In this section, we will demonstrate equivalence between the Quantum Inference Protocol's inferred quantum memory states and the exact quantum memory states. The latter are obtained through an ideal ε -machine if the chosen L is at least the Markov order of the process.

The Markov order R is defined as the minimum length of sequence R where $P(\vec{x}|x_{-\infty:0}) = P(\vec{x}|x_{-R:0})$ [59]. Since we map pasts to causal states, it makes sense to denote the Markov order as the smallest history length r to determine the causal state of a process [21],

$$R = \min\{r : H(S_0|x_{-r:0}) = 0\} \quad (4.5)$$

Suppose we had two inferred quantum memory states $|s_{x_{-L:0}}\rangle$ and $|s_{x'_{-L:0}}\rangle$ that are defined by two pasts $x_{-L:0}$ and $x'_{-L:0}$,

$$\begin{aligned} U|s_{x_{-L:0}}\rangle|0\rangle &= \sum_{x_0} \sqrt{\tilde{P}(x_0|x_{-L:0})} |s_{x_{-L+1:1}}\rangle |x_0\rangle \\ U|s_{x'_{-L:0}}\rangle|0\rangle &= \sum_{x'_0} \sqrt{\tilde{P}(x'_0|x'_{-L:0})} |s_{x'_{-L+1:1}}\rangle |x'_0\rangle. \end{aligned} \quad (4.6)$$

Taking their inner products, we have

$$\begin{aligned} \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle &= \langle s_{x_{-L:0}} | U^\dagger U | s_{x'_{-L:0}} \rangle \\ &= \sum_{x_0, x'_0} \sqrt{\tilde{P}(x_0|x_{-L:0}) \tilde{P}(x'_0|x'_{-L:0})} \langle s_{x_{-L+1:1}} | s_{x'_{-L+1:1}} \rangle \langle x_0 | x'_0 \rangle. \end{aligned} \quad (4.7)$$

Because $\langle x_0 | x'_0 \rangle = \delta_{x_{-L:0} x'_{-L:0}}$ and $U^\dagger U = 1$, the inner products are reduced to

$$\langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle = \sum_{x_0} \sqrt{\tilde{P}(x_0 | x_{-L:0}) \tilde{P}(x_0 | x'_{-L:0})} \langle s_{x_{-L+1:1}} | s_{x'_{-L+1:1}} \rangle. \quad (4.8)$$

By iteratively working out the inner products between $\langle s_{x_{-L+1:1}} | s_{x'_{-L+1:1}} \rangle$ and so on, we arrive at a recurrence relation,

$$\begin{aligned} \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle &= \sqrt{\tilde{P}(x_0 | x_{-L:0}) \tilde{P}(x_0 | x'_{-L:0})} \times \\ &\quad \sqrt{\tilde{P}(x_1 | x_{-L+1:1}) \tilde{P}(x_1 | x'_{-L+1:1})} \times \\ &\quad \dots \\ &= \sum_{x_0, x_1, \dots} \sqrt{\tilde{P}(x_0 x_1 \dots | x_{-L:0}) \tilde{P}(x_0 x_1 \dots | x'_{-L:0})} \\ &= \sum_{\vec{x}} \sqrt{\tilde{P}(\vec{x} | x_{-L:0}) \tilde{P}(\vec{x} | x'_{-L:0})}. \end{aligned} \quad (4.9)$$

Given that $L \geq R$, the past $x_{-L:0}$ maps to the exact causal state of $\overleftarrow{x}$, the full pasts $\overleftarrow{x}$ and $\overleftarrow{x}'$ can be represented by any L -length past, giving the following equation

$$\langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle = \sum_{\vec{x}} \sqrt{\tilde{P}(\vec{x} | \overleftarrow{x}) \tilde{P}(\vec{x} | \overleftarrow{x}')}. \quad (4.10)$$

We can conclude that the inner products $\langle s_{x_{0:L}} | s_{x'_{0:L}} \rangle$ can be reduced to

$$\langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle = 1 \quad (4.11)$$

if and only if the condition holds,

$$\tilde{P}(\vec{X} | \overleftarrow{x}) = \tilde{P}(\vec{X} | \overleftarrow{x}'). \quad (4.12)$$

This implies that if $\tilde{P}(\vec{X} | \overleftarrow{x})$ is equal to $\tilde{P}(\vec{X} | \overleftarrow{x}')$, two quantum memory states $|s_{x_{-L:0}}\rangle$ and $|s_{x'_{-L:0}}\rangle$ are equivalent. Their inner products are 1. This completes the proof that

the inference protocol's quantum memory states and the exact quantum memory states are equivalent when $L \geq R$ for $x_{-L:0} \sim x'_{-L:0}$, allowing self-merging to occur at and above the Markov order.

4.4 Robustness of C_q

To prove the robustness of the quantum statistical memory, C_q , we will allow probabilities to be perturbed as follows

$$P(\vec{X}) \rightarrow P^\epsilon(\vec{X}) = P(\vec{X}) + \epsilon \Delta P(\vec{X}), \quad (4.13)$$

where ϵ represents the strength of the perturbation and $\Delta P(\vec{X})$ represents the relative changes in distribution for each string $\vec{X}$.

Although C_q is traditionally computed by taking $-\text{tr}(\rho \log_2 \rho)$, we will make use of the Gram matrix to prove the robustness of C_q . The Gram matrix G is a matrix of rescaled inner products of quantum memory states [22, 60, 61] that has the same spectrum as ρ and can be used to efficiently show the effects of perturbations to the probabilities. It was introduced in Ref. [60] to compare two ensembles with the same number of quantum states with same probabilities and used in Ref. [22] to compute C_q with greater efficiency.

The elements of the Gram matrix are obtained by taking overlaps between quantum states together with a product of their probabilities. It has elements given by $G_{jk} = \sqrt{P_j P_k} \langle \sigma_j | \sigma_k \rangle$. We will apply the Gram matrix to our model where the quantum memory states are denoted by $|\varsigma_{x_{-L:0}}\rangle$ and $|\varsigma_{x'_{-L:0}}\rangle$. Assuming the quantum memory states have exact probabilities, the elements of the Gram matrix are

$$G_{x_{-L:0} x'_{-L:0}} = \sqrt{P(x_{-L:0}) P(x'_{-L:0})} \langle \varsigma_{x_{-L:0}} | \varsigma_{x'_{-L:0}} \rangle. \quad (4.14)$$

The overlaps can be written with Eq. (4.10) but we consider the case for L -length pasts and futures when $L \geq R$,

$$\langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle = \sum_{x_{0:L}} \sqrt{P(x_{0:L} | x_{-L:0}) P(x_{0:L} | x'_{-L:0})}. \quad (4.15)$$

Substituting back to Eq. (4.14), we obtain

$$\begin{aligned} G_{x_{-L:0} x'_{-L:0}} &= \sum_{x_{0:L}} \sqrt{P(x_{-L:0}) P(x'_{-L:0})} \sqrt{P(x_{0:L} | x_{-L:0}) P(x_{0:L} | x'_{-L:0})} \\ &= \sum_{x_{0:L}} \sqrt{P(x_{-L:0} x_{0:L}) P(x'_{-L:0} x_{0:L})}. \end{aligned} \quad (4.16)$$

We first work out how the perturbation affects these probabilities

$$\sqrt{P(x_{-L:0} x_{0:L}) P(x'_{-L:0} x_{0:L})}$$

before showing how the perturbation affects the Gram matrix G . The square-root of two arbitrary perturbed probabilities P^ϵ and Q^ϵ multiplied together is found to first order of ϵ as

$$\begin{aligned} \sqrt{P^\epsilon Q^\epsilon} &= \sqrt{(P + \epsilon \Delta P)(Q + \epsilon \Delta Q)} \\ &= \sqrt{PQ + P\epsilon \Delta Q + Q\epsilon \Delta P + (\epsilon \Delta P)(\epsilon \Delta Q)} \\ &= \sqrt{PQ} \sqrt{1 + \frac{\epsilon \Delta Q}{Q} + \frac{\epsilon \Delta P}{P} + \frac{\epsilon^2 \Delta P \Delta Q}{PQ}} \\ &= \sqrt{PQ} \left[1 + \frac{1}{2} \left(\frac{\epsilon \Delta Q}{Q} + \frac{\epsilon \Delta P}{P} + \frac{\epsilon^2 \Delta P \Delta Q}{PQ} \right) + O(\epsilon^3) + \dots \right] \\ &\approx \sqrt{PQ} + \frac{\sqrt{PQ}}{2} \left(\frac{\epsilon \Delta P}{P} + \frac{\epsilon \Delta Q}{Q} \right) + O(\epsilon^2). \end{aligned} \quad (4.17)$$

We now substitute Eq. (4.17) into Eq. (4.14) to see how perturbations to the probabilities affects the Gram matrix. The result is that we have a perturbed Gram

matrix with the following elements

$$\begin{aligned}
 G_{x_{-L:0}x'_{-L:0}}^\epsilon &= \sum_{x_{0:L}} \sqrt{P^\epsilon(x_{-L:0}x_{0:L})P^\epsilon(x'_{-L:0}x_{0:L})} \\
 &\approx \sum_{x_{0:L}} \sqrt{P(x_{-L:0}x_{0:L})P(x'_{-L:0}x_{0:L})} + \\
 &\quad \frac{\epsilon \sqrt{P(x_{-L:0}x_{0:L})P(x'_{-L:0}x_{0:L})}}{2} \left(\frac{\Delta P(x_{-L:0}x_{0:L})}{P(x_{-L:0}x_{0:L})} + \frac{\Delta P(x'_{-L:0}x_{0:L})}{P(x'_{-L:0}x_{0:L})} \right) + \\
 &\quad O(\epsilon^2).
 \end{aligned} \tag{4.18}$$

The perturbed Gram matrix is then shown to be separated into two components with the exact and the perturbation term. Each element of the Gram matrix is $G_{x_{-L:0}x'_{-L:0}}^\epsilon = G_{x_{-L:0}x'_{-L:0}} + \epsilon \Delta G_{x_{-L:0}x'_{-L:0}}$. To first order of ϵ , we have the components of G^ϵ given as

$$\begin{aligned}
 G_{x_{-L:0}x'_{-L:0}} &= \sum_{x_{0:L}} \sqrt{P(x_{-L:0}x_{0:L})P(x'_{-L:0}x_{0:L})} \\
 &\text{and} \\
 \epsilon \Delta G_{x_{-L:0}x'_{-L:0}} &= \sum_{x_{0:L}} \frac{\epsilon \sqrt{P(x_{-L:0}x_{0:L})P(x'_{-L:0}x_{0:L})}}{2} \left(\frac{\Delta P(x_{-L:0}x_{0:L})}{P(x_{-L:0}x_{0:L})} + \frac{\Delta P(x'_{-L:0}x_{0:L})}{P(x'_{-L:0}x_{0:L})} \right)
 \end{aligned} \tag{4.19}$$

From Weyl's theorem for perturbed Hermitian matrices, the perturbation to the eigenvalues of G^ϵ is upper bounded by largest the eigenvalue of $\epsilon \Delta G$ [61]. Adjusting the scaling parameter to zero, $\epsilon \rightarrow 0$, G^ϵ smoothly converges to G . Now, since the von Neumann entropy of a density matrix is continuous on its spectrum [44], and the density matrix and the Gram matrix share the same spectrum, the von Neumann entropy is also continuous on the spectrum of the Gram matrix. Therefore, the von Neumann entropy smoothly varies with respect to ϵ , proving that the quantum statistical memory is robust with respect to ϵ .

By viewing the same problem from a different lens, we observe how adding scaling

parameter to the density matrix, ρ , results in the scaling for $C_q = -\text{tr}(\rho \log_2 \rho)$. By allowing $\rho^\epsilon = \rho + \epsilon \Delta \rho$, the perturbed von Neumann entropy C_q^ϵ becomes

$$\begin{aligned}
 C_q^\epsilon &= -\text{tr}(\rho^\epsilon \log_2 \rho^\epsilon) \\
 &= -\text{tr}[(\rho + \epsilon \Delta \rho) \log_2(\rho + \epsilon \Delta \rho)] \\
 &= -\text{tr}[\rho \log_2(\rho + \epsilon \Delta \rho) + \epsilon \Delta \rho \log_2(\rho + \epsilon \Delta \rho)] \\
 &= -\text{tr}\left[\rho \left(\log_2 \rho + \log_2 \left(1 + \frac{\epsilon \Delta \rho}{\rho}\right)\right) + \epsilon \Delta \rho \left(\log_2 \rho + \log_2 \left(1 + \frac{\epsilon \Delta \rho}{\rho}\right)\right)\right].
 \end{aligned} \tag{4.20}$$

Given the Taylor expansion for $\log_e(1+x)$ is

$$\log_e(1+x) = x - (1/2)x^2 + (1/3)x^3 + \dots \tag{4.21}$$

and $\log_2(1+x) = \log_e(1+x)/\log_e(2)$, we have—to first order of ϵ —the following solution for C_q^ϵ ,

$$\begin{aligned}
 C_q^\epsilon &= -\text{tr}\left[\rho \left(\log_2 \rho + \frac{1}{\ln 2} \left(\frac{\epsilon \Delta \rho}{\rho}\right)\right) + \epsilon \Delta \rho \left(\log_2 \rho + \frac{1}{\ln 2} \left(\frac{\epsilon \Delta \rho}{\rho}\right)\right)\right] \\
 &= -\text{tr}\left[\rho \log_2 \rho + \frac{\epsilon \Delta \rho}{\ln 2} + \epsilon \Delta \rho \log_2 \rho + \frac{(\epsilon \Delta \rho)^2}{\rho \ln 2} + \dots\right] \\
 &\approx -\text{tr}(\rho \log_2 \rho) - \text{tr}\left[\epsilon \Delta \rho \left(\log_2 \rho + \frac{1}{\ln 2}\right)\right] \\
 &= C_q - \epsilon \text{tr}\left[\Delta \rho \left(\log_2 \rho + \frac{1}{\ln 2}\right)\right].
 \end{aligned} \tag{4.22}$$

As such, it is shown that when $\epsilon \rightarrow 0$, $C_q^\epsilon \rightarrow C_q$ smoothly.

4.5 The Inference Protocol

By establishing that (1) quantum memory states undergo self-merging at and above the Markov order and (2) the robustness of C_q to perturbations in the probability distributions, we can now argue the accuracy of the inference protocol. By parsing through a given stochastic process with the frequentist approach, the marginal

distributions of $(L + 1)$ -length words can be inferred (for some choice of L). The marginal distributions are used to estimate the conditional probabilities in Eq. (4.1) by concatenation. Moreover, the marginal distributions can also be used to construct estimated conditional probabilities for Eq. (4.4). The estimates for the conditional distributions are evaluated as

$$\tilde{P}(x_{0:L}|x_{-L:0}) = \tilde{P}(x_0|x_{-L:0})\tilde{P}(x_1|x_{-L+1:1})\dots\tilde{P}(x_{L-1}|x_{-L+L:L}). \quad (4.23)$$

Eq. (4.23) should result in an estimate that is close to the exact distributions provided the marginal distributions are close to the exact marginal distributions, and that $L \geq R$.

Finite data would result in statistical fluctuations in the marginal distributions. This should not affect the methodology if the fluctuations are small. These statistical fluctuations can be treated as a perturbation to the probabilities shown in Sec. 4.4 but with ΔP being stochastic variables. We show in Sec. 4.6 that the size of error in the inferred Gram matrix and estimated quantum statistical memory scales as $O\left(\frac{|A|^L}{\sqrt{N}}\right)$ where N is the length of the stochastic process (the size of data).

The next question is how we determine the choice of L . We recommend that the maximum choice of L , given by $L_{\max}$, be not more than $L_{\max} \lesssim \log_{|A|}(N/1,000)$. Too large an L will lead to large fluctuations in the probability distributions because of undersampling. However, we also require L to be large enough for the Markov order of the process for a good estimate of $\tilde{C}_q$.

For a process that has infinite Markov order, it would be computationally infeasible to have $L_{\max} \rightarrow \infty$. While knowing the Markov order rightfully indicates how far back into the past one should look into, it does not provide the strength of these long-range correlations. Typically, these distant pasts can be treated as small perturbations to the statistics to the most recent past and so we can expect minimal impact from

these perturbations.

We introduce the concept of the effective Markov order R_{eff} to promote the idea that having a shorter past than the Markov order may still be ‘good enough’ for capturing most of the predictive information in the past. The effective Markov order is defined as the minimum length of observations for which having an additional time step into the past does not exceed some threshold tolerance value δ ,

$$R_{\text{eff}} = \min\{r : \max_{xx'} \langle D(P(X_0|xX_{-r:0}), P(X_0|x'X_{-r:0})) \rangle < \delta\}. \quad (4.24)$$

Here, $D(P, Q)$ is some measure of distance between two probability distributions P and Q . For the purposes of this work, we use the trace distance measure

$$D_{\text{TD}}(P, Q) = \frac{1}{2} \sum_x |P(x) - Q(x)|. \quad (4.25)$$

4.6 Scaling of the error

A finite-length stochastic process begets imperfect probability distributions analogously to a finite number of flips for a fair coin that gives $P(\text{head}) \approx 0.5$. The approximation of probabilities can be treated as a perturbation to the exact probabilities, $\tilde{P} = P + \Delta P$. How does the error scale with the amount of statistical noise that arises from finite data?

Suppose the stochastic process is of length N . With L -length words, the inference protocol will estimate $\tilde{C}_q$ of the process through building up a functional set of quantum memory states constructed with L -length words. How will the error to the Gram matrix scale? Using results from Eq. (4.19), we see that the elements of the

inferred Gram matrix have the following error,

$$\Delta G_{x_{-L:0}x'_{-L:0}} = \sum_{x_{0:L}} \frac{\sqrt{P(x_{-L:0}x_{0:L})P(x'_{-L:0}x_{0:L})}}{2} \left(\frac{\Delta P(x_{-L:0}x_{0:L})}{P(x_{-L:0}x_{0:L})} + \frac{\Delta P(x'_{-L:0}x_{0:L})}{P(x'_{-L:0}x_{0:L})} \right). \quad (4.26)$$

Instead of having ϵ as the scaling of the perturbation, the strength of perturbation is instead dictated by the uncertainty terms (given with the Δ) in Eq. (4.26). For a randomly selected L -length word, it is either $x_{-L:0}$ or not. Thus the probability distribution $P(x_{-L:0})$ follows a binomial distribution and can be approximated to a normal distribution when N is large, where the standard deviation is $\sigma = \sqrt{P(x_{-L:L})(1 - P(x_{-L:L}))}$. Since the exact distribution is not known, the standard error of the mean of the distribution is used to estimate the error. In general, the standard error of the mean of an arbitrary distribution with N amounts of data is $\sigma_{\text{mean}} = \sigma/\sqrt{N}$. Similarly, the standard error of the mean for $P(x_{-L:0})$ is given by

$$\sigma_{P(x_{-L:L})} \equiv \Delta P(x_{-L:L}) = \sqrt{\frac{P(x_{-L:L})(1 - P(x_{-L:L}))}{N}}. \quad (4.27)$$

Substituting Eq. (4.27) into Eq. (4.26), we obtain

$$\begin{aligned} \Delta G_{x_{-L:0}x'_{-L:0}} &= \frac{1}{2\sqrt{N}} \sum_{x_{0:L}} \sqrt{P(x'_{-L:0}x_{0:L})(1 - P(x_{-L:0}x_{0:L}))} \\ &\quad + \sqrt{P(x_{-L:0}x_{0:L})(1 - P(x'_{-L:0}x_{0:L}))}. \end{aligned} \quad (4.28)$$

The probability of obtaining an L -length sequence decreases approximately exponentially with L . Assuming that all L -length words (for long L) are almost uniformly distributed, $P(x_{-L:L})$ occurs with probability $1/|\mathcal{A}|^{2L} = |\mathcal{A}|^{-2L}$. Considering that there are roughly $|\mathcal{A}|^L$ terms in the sum, we have $\Delta G_{x_{-L:0}x'_{-L:0}} \sim 1/\sqrt{N}$. Finally, because the dimension of the matrix scales as $|\mathcal{A}|^L$, hence the spectral norm of $\Delta G_{x_{-L:0}x'_{-L:0}}$ would

generally scale as

$$\Delta G_{X_{-L:0}X'_{-L:0}} \propto O\left(\frac{|\mathcal{A}|^L}{\sqrt{N}}\right). \quad (4.29)$$

4.7 Examples

Two examples of stochastic processes that are generated by two generators are used to demonstrate the capabilities of the Quantum Inference Protocol of estimating $\tilde{C}_q$ [25]. Since both generators are in fact explicit ε -machines, one can work out the exact analytical solution for C_q to use for benchmarking. The first generator is the 3-2 Golden Mean Process, which has a finite Markov order of $R = 3$ and cryptic order of $k = 2$ (Fig. 4.4(a)). The second generator is the Nemo Process. The Nemo Process is a process with an infinite Markov order (Fig. 4.3(a)).

3-2 Golden Mean Process. The first example is given by a ε -machine obtained from the R - k Golden Mean Process family. R and k represent the Markov order and cryptic order respectively which are tunable parameters. We consider the 3-2 Golden Mean Process in Fig. 4.3(a).

In Fig. 4.3(b), we observe that the expectation of the trace distance between differing symbols increases far into the past between $\tilde{P}(X_0|0X_{-L:0})$ and $\tilde{P}(X_0|1X_{-L:0})$ from which we can infer the effective Markov order defined in Eq. (4.24). The limits of finite data are visible when L is too large relative to N ; undersampling of the process statistics causes $D_{TD}(\tilde{P}(X_0|0X_{-R:0}), \tilde{P}(X_0|1X_{-R:0}))$ to fluctuate. The hollow circle on each plot represents the point at which $L = \log_2(N/1000)$, beyond this point we consider statistics to be undersampled. Setting a threshold $\delta = 0.01$, we would assign an effective Markov order of $R_{\text{eff}} = 3$ which aligns with the exact Markov order of the process.

Figure 4.3: (a) 3-2 Golden Mean Process with $p = 0.9$. (b) Trace distance of $(\tilde{P}(X_0|0X_{-L:0})$ and $(\tilde{P}(X_0|1X_{-L:0})$ vs L . (c) $\tilde{C}_q(L)$ vs L .

Figure 4.4: (a) Nemo Process set to $p = 0.1$ and $q = 0.9$. (b) Trace distance of $(\tilde{P}(X_0|0X_{-L:0})$ and $(\tilde{P}(X_0|1X_{-L:0})$ vs L . (c) $\tilde{C}_q(L)$ vs L .

Nemo Process. The second example is given by the Nemo Process, whose ε -machine is represented in Fig. 4.4(a).

The Nemo process features an infinite Markov order as seen by having a contiguous string of zeros of any length that cannot be exactly synchronised to any causal state. This implies the impossibility of choosing an L that matches the Markov order of the process. This is where the effective Markov order will be useful and will provide a suitable proxy for the Markov order.

Similarly, Fig. 4.4(b) shows the expected trace distance between $\tilde{P}(X_0|0X_{-L:0})$ and $\tilde{P}(X_0|1X_{-L:0})$. Setting a tolerance of $\delta = 0.01$, we observe that the trace distance remains below the delta tolerance at $L \geq 6$ (especially for large N). As such, we can assign the effective Markov order to be $R_{\text{eff}} = 6$.

With $L = 6$, we see that the estimated quantum statistical memory is accurate, $\tilde{C}_q \approx C_q$. This strikes a balance between allowing sufficiently long histories to capture most of the past dependency while not underfitting the probability distributions. Notice that for $N = 10^4$ illustrates the effects of insufficient data to provide a good estimate; the L_{max} corresponding to $N = 10^4$ is significantly smaller than R_{eff} and can be deduced from the trace distance in Fig. 4.4(b), which never drops below the delta tolerance.

4.8 Summary

In this Chapter, we have introduced the Quantum Inference Protocol, which is an inference algorithm that has dual purposes. The Quantum Inference Protocol can be used to construct quantum memory states without the need for mapping a ϵ -machine's causal states to quantum states. This reduces any errors that may be incurred due to the δ -tolerance used to construct causal states. The Quantum Inference Protocol is also able to quickly identify an estimate for the memory cost of simulating stochastic processes with quantum causal models.

The validity of the inferred quantum memory states was justified by first showing that the inferred quantum memory states undergo self-merging when the length of consecutive past outputs is at or greater than the Markov order of the process. We then showed that the quantum statistical memory is robust to perturbations. The robustness was shown by allowing probabilities to be perturbed with a scaling factor. The above-mentioned two points set the stage for the Quantum Inference Protocol.

As long as the choice of consecutive pasts to observe is at or greater than the Markov order, the inferred quantum statistical memory will be close to the estimate, but still being dependent on the amount of data given. The inferred quantum statistical memory is robust against statistical perturbations arising from finite data.

Summary for the Quantum Inference Protocol

The Quantum Inference Protocol was developed to remove reliance on classical causal models to obtain quantum causal models of stochastic processes. Classical causal models require a user-defined tolerance value to compare probability distributions, causing them to be extremely sensitive to perturbations in the stochastic process.

The Quantum Inference Protocol infers the same stochastic process and builds quantum states that satisfy the following equation

$$U|\varsigma_{x_{-L:0}}\rangle|0\rangle = \sum_{x_0 \in \mathcal{A}} \sqrt{\tilde{P}(x_0|x_{-L:0})} |\varsigma_{x_{-L+1:1}}\rangle |x_0\rangle. \quad (4.30)$$

and its corresponding quantum statistical memory is found by

$$\tilde{C}_q(L) = -\text{tr}(\rho(L) \log_2 \rho(L)), \quad (4.31)$$

with $\rho(L) = \sum_{x_{-L:0}} \tilde{P}(x_{-L:0}) |\varsigma_{x_{-L:0}}\rangle \langle \varsigma_{x_{-L:0}}|$ being the density matrix.

For a quick gauge on the amount of memory quantum causal models require, one may use a model-independent method to obtain the inferred Gram matrix as the Gram matrix and density matrix have the same spectrum. The Gram matrix has elements

$$G_{x_{-L:0}x'_{-L:0}} = \sum_{x_{0:L}} \sqrt{\tilde{P}(x_{-L:0}x_{0:L})\tilde{P}(x'_{-L:0}x_{0:L})}. \quad (4.32)$$

We showed that $\tilde{C}_q \approx C_q$ if $L \geq R$ and if statistical fluctuations to the marginal distributions are small. The choice of L is recommended to be upper bounded at $L_{\max} \lesssim \log_{|\mathcal{A}|}(N/1,000)$.

CHAPTER 5

Unitary construction and decomposition

5.1 Merging quantum memory states

The methodology for constructing the unitary for any given set of quantum memory states also applies to the set of inferred quantum memory states obtained with the Quantum Inference Protocol.

The Quantum Inference Protocol produces a maximum of $|\mathcal{A}|^L$ quantum memory states. The ensuing unitary operator would also have dimensions $|\mathcal{A}|^L$ by $|\mathcal{A}|^L$. Constructing a unitary operator of size $|\mathcal{A}|^L$ by $|\mathcal{A}|^L$ is theoretically possible but practically infeasible. A large matrix tends to have large circuit depth after decomposing into elementary quantum gates that are implementable on a quantum device. A large circuit depth tends to incur more noise.

Therefore, we strongly recommend merging together statistically similar quantum memory states up to a δ tolerance to reduce the state space such that the unitary operator is within implementable means. Reducing the dimensions of the unitary operator ensures that the number of qubits needed is within acceptable and imple-

mentable means. ‘Statistical similarity’ of quantum states can be found with the fidelity measure.

We compute the fidelity between two inferred quantum memory state, $|\varsigma_{x-L:0}\rangle$ and $|\varsigma_{x'-L:0}\rangle$, essentially invoking an alternate version of the equivalence relation. If the fidelity is greater than $1 - \delta$, the two inferred quantum memory states are deemed to be statistically similar and merged into a single quantum memory states and given a label $|\varsigma_j\rangle$.

$$|\varsigma_{x-L:0}\rangle \sim |\varsigma_{x'-L:0}\rangle \iff |\langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle| \geq 1 - \delta. \quad (5.1)$$

As the accuracy of the probability amplitudes of the inferred quantum memory states depends highly on the length of stochastic process N , we expect δ to be dependent on N . We set $\delta = \frac{1}{2\sqrt{N}}$ with the derivation found in Section 5.4. One has freedom to demand a stricter criteria for statistical similarity by having $\delta < \frac{1}{2\sqrt{N}}$.

Invoking Eq. (5.1) then gives the set of merged inferred quantum memory states $\{|\varsigma_j\rangle\}$. We can then proceed with constructing the unitary operator for the set of merged quantum memory states.

5.2 Building the unitary

The procedure for constructing the unitary operator for any given set of quantum memory states, be it inferred using the Quantum Inference Protocol or obtained via a direct mapping from the causal states of an ε -machine consists of four main steps [23].

- (1) Compute the inner products between two quantum memory states, $c_{ij} = \langle \sigma_i | \sigma_j \rangle$.
- (2) With an inverse Gram-Schmidt procedure, express the set of quantum memory states $\{|\sigma_j\rangle\}$ in a chosen orthogonal basis $\{|e_j\rangle\}$ in Hilbert space of the

appropriate dimensions. This gives the following set of equations

$$\begin{aligned} |\sigma_1\rangle &= |e_1\rangle \\ |\sigma_2\rangle &= c_{12}|e_1\rangle + \sqrt{1 - c_{12}^2}|e_2\rangle, \text{ etc.} \end{aligned} \quad (5.2)$$

(3) Solve for matrix elements

$$U_{ix,i'0} = \langle e_i | \langle x | U | e_{i'} \rangle | x = 0 \rangle. \quad (5.3)$$

The $|x = 0\rangle$ term indicates that Eq. (5.3) will fill in the odd-numbered columns of the unitary operator.

(4) Fill in the remaining even-numbered columns of U with the Gram-Schmidt procedure taking into account the already-determined columns in Step (3). As a straightforward example, random numbers can be assigned to these even-numbered columns before invoking the Gram-Schmidt procedure.

Given the set of merged inferred quantum memory states $\{|s_i\rangle\}$, one may write the basis states $|e_i\rangle$ as a linear combination of the merged quantum states.

$$\begin{aligned} |e_{i=1}\rangle &= \gamma_{i=1,k=1} |s_{k=1}\rangle \\ |e_{i=2}\rangle &= \gamma_{i=2,k=1} |s_{k=1}\rangle + \gamma_{i=2,k=2} |s_{k=2}\rangle \\ |e_{i=3}\rangle &= \gamma_{i=3,k=1} |s_{k=1}\rangle + \gamma_{i=3,k=2} |s_{k=2}\rangle + \gamma_{i=3,k=3} |s_{k=3}\rangle \end{aligned} \quad (5.4)$$

with $\gamma_{i,k}$ being the respective coefficients that can be found with the Gram-Schmidt procedure. Substituting Eq. (5.4) into Eq. (5.3), each element within the unique columns of the unitary operator can be found as

$$U_{ix,i'0} = \sum_{k=1}^{k=i} \sum_{k'=1}^{k'=i'} (\gamma_{i,k}) (\gamma_{i',k'}) \sqrt{P(x|k')} \langle s_i | s_{\lambda(x,k')} \rangle. \quad (5.5)$$

In our algorithm, we assign random numbers to the remaining even-numbered columns before orthonormalising them with the columns obtained in Eq. (5.5).

The resultant quantum memory states for which the unitary operator acts requires only that the overlaps $\langle s_i | s_j \rangle$ be preserved. The merged quantum memory states can then be written in column form as

$$\begin{aligned} |s_1\rangle &= [1, 0, 0\dots]^T, \\ |s_2\rangle &= [\langle s_1 | s_2 \rangle, \sqrt{1 - \langle s_1 | s_2 \rangle^2}, 0\dots]^T \text{ etc.} \end{aligned} \quad (5.6)$$

5.3 Decomposing the unitary

Any operator has to be decomposed into elementary quantum gates that are implementable on a quantum device. Examples of elementary quantum gates include the standard single qubit Pauli gates, Hadamard gate, rotation gates around the X, Y, and Z directions, controlled-NOT, Toffoli gate, etc. There exists several decomposition methods [62–69], of which some have been automated with codes. Although Qiskit—the platform we use for numerical work—features its own version of two qubit unitary decomposition [70], we opt not to use it for two main reasons. It is limited to two qubits, but more importantly, our operator is not guaranteed to be a numerically exact operator where $U^\dagger U = I$. This is due to statistical fluctuations from the finite stochastic process affecting the probability amplitudes of quantum memory states, which in turn affects the relevant constructed unitary operator.

Therefore, in our code, we will make use of the cosine-sine decomposition [64] coupled with the generalised singular value decomposition [62].

The cosine-sine decomposition states that any $2^n \times 2^n$ matrix can be split into four block matrices $g_{11}, g_{12}, g_{21}, g_{22}$ of size $2^{n-1} \times 2^{n-1}$. Singular value decomposition is then applied to any two of these four block matrices to obtain the following decomposition,

Figure 5.1: The unitary is decomposed into elementary quantum gates comprising of CNOTs, controlled- Z rotations, controlled- Y rotations, controlled-phase gates. We labelled u_1 , u_2 , v_1 , and v_2 to represent the segments according to Eq. (5.7). The multiplexor is made up of diagonal cosine and sine terms (see Eq. (5.9)).

$$U \equiv \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} u_1 & \\ & u_2 \end{bmatrix} \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}. \quad (5.7)$$

where

$$\begin{aligned} g_{11} &= u_1 c v_1; & g_{12} &= -u_1 s v_2; \\ g_{21} &= u_2 s v_1; & g_{22} &= u_2 c v_2. \end{aligned} \quad (5.8)$$

Here and elsewhere, spaces in matrices are square matrices consisting of all zero entries of the appropriate dimensions.

The middle term $\begin{bmatrix} c & -s \\ s & c \end{bmatrix}$ in Equation (5.7) is what gives the cosine-sine decomposition its name as it constitutes square matrices with diagonal cosine and sine terms as follows,

$$c \equiv \begin{bmatrix} \cos \theta_1 & & & \\ & \cos \theta_2 & & \\ & & \ddots & \\ & & & \cos \theta_{2^{n-1}} \end{bmatrix}; \quad s \equiv \begin{bmatrix} \sin \theta_1 & & & \\ & \sin \theta_2 & & \\ & & \ddots & \\ & & & \sin \theta_{2^{n-1}} \end{bmatrix}. \quad (5.9)$$

This middle term is also known as a multiplexor gate [65]. A multiplexor gate is one which allows ‘if-else’ conditions. An example of a multiplexor gate is the CNOT gate. If the control qubit is $|0\rangle$, nothing happens to the target qubit. If the control qubit is $|1\rangle$, then the state of the target qubit is flipped from $|0\rangle$ to $|1\rangle$ or vice versa.

The simplest of the cosine-sine multiplexor gate contains two angles θ_1 and θ_2 where a Pauli Y -rotation with angle θ_1 is applied to the first qubit (target qubit) if the second qubit (control qubit) is $|0\rangle$ and a Y -rotation with angle θ_2 is applied to the first qubit if the second qubit is $|1\rangle$, i.e.

$$\begin{bmatrix} c & -s \\ s & c \end{bmatrix} |q_0\rangle |q_1\rangle = \begin{cases} (R_y(\theta_1) \otimes I) |q_0\rangle |q_1\rangle, & \text{if } |q_1\rangle = |0\rangle \\ (R_y(\theta_2) \otimes I) |q_0\rangle |q_1\rangle, & \text{if } |q_1\rangle = |1\rangle. \end{cases} \quad (5.10)$$

To further simplify the matrices $\begin{bmatrix} u_1 & \\ & u_2 \end{bmatrix}$ and $\begin{bmatrix} v_1 & \\ & v_2 \end{bmatrix}$ in Equation (5.7), we use the matrix identity [71]

$$\begin{bmatrix} \alpha & \\ & \beta \end{bmatrix} = \begin{bmatrix} I & \\ & I \end{bmatrix} \begin{bmatrix} I & \\ & \alpha \end{bmatrix} \begin{bmatrix} I & \\ & I \end{bmatrix} \begin{bmatrix} I & \\ & \beta \end{bmatrix}, \quad (5.11)$$

with $\alpha, \beta \in U(2^{n-1})$ and I is a $2^{n-1} \times 2^{n-1}$ identity matrix. Furthermore,

$$\begin{bmatrix} & I \\ I & \end{bmatrix} \equiv X \otimes I \quad (5.12)$$

with X being NOT gate. The matrix $\begin{bmatrix} I & \\ & \alpha \end{bmatrix}$ is a controlled- α gate where α is unitary.

The left and right matrices $\begin{bmatrix} u_1 & \\ & u_2 \end{bmatrix}$ and $\begin{bmatrix} v_1 & \\ & v_2 \end{bmatrix}$ are simplified by repeatedly applying singular value decomposition to u_1, u_2, v_1, v_2 until $U(2)$ is achieved. Once the single qubit $U(2)$ matrices are obtained, they can be decomposed into either ZYZ (or ZXZ) rotations with some angles $\gamma, \phi, \theta, \lambda$ [44]

$$U(2) = e^{i\gamma} R_Z(\phi) R_Y(\theta) R_Z(\lambda). \quad (5.13)$$

The circuit depth of the cosine-sine decomposition method for a $2^n \times 2^n$ unitary operator generally scales as

$$[7(4^{n-1}) + 5(\sum_{i=0}^{n-2} 4^i)]. \quad (5.14)$$

As a proof of concept, we aim to apply this methodology to decompose a 4×4 unitary operator that consists of two quantum memory states. The decomposition is illustrated in Fig. 5.2.

The algorithms for the construction and decomposition of the the unitary operator has been made available on GitHub at <https://github.com/matthew0021/unitary-construction-and-decomposition>.

The illustration for a unitary operator acting on $|\sigma_j\rangle|0\rangle$ (or equivalently $|\varsigma_{x-L,0}\rangle|0\rangle$) is given in Fig. 5.2(a). $|\sigma_j\rangle$ forms the memory register and $|0\rangle$ is the output register.

In the circuit representation, if $|\sigma_j\rangle|0\rangle$ forms a system of two qubits, the memory qubit functions as the control qubit while new blank ancillary qubits $|0\rangle$ are used for every unitary operation. The circuit representation is depicted in Fig. 5.2(b). Lastly, Fig. 5.2(c) shows how a two-qubit unitary operator is decomposed into a series of controlled-rotations, multiplexors, and single qubit gates.

5.4 δ -similarity of quantum memory states

Fidelity is used as a measure of similarity between two quantum memory states $|\varsigma_{x-L:0}\rangle$ and $|\varsigma_{x'-L:0}\rangle$. It is defined as $|\langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle|$. If the fidelity is 1, it implies that the two quantum memory states are identical; if it is zero, then the two quantum memory states are fully orthogonal.

The reason for having a δ -tolerance is because a stochastic process of finite length N incurs imperfect probability amplitudes. These probability amplitudes will tend towards the exact probability amplitudes with increasing amounts of N . Since the quantum memory states have real and positive probability amplitudes, the absolute sign is not necessary. The fidelity defines the following equivalence relation

$$|\varsigma_{x-L:0}\rangle \sim |\varsigma_{x'-L:0}\rangle \iff |\langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle| \geq 1 - \delta. \quad (5.15)$$

We solve for the inner products $\langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle$ to observe how fidelity scales with N .

$$\begin{aligned} \langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle &= \langle \varsigma_{x-L:0} | U^\dagger U | \varsigma_{x'-L:0} \rangle \\ &= \sum_{x_0 \in \mathcal{A}} \sqrt{\tilde{P}(x_0 | x_{-L:0}) \tilde{P}(x_0 | x'_{-L:0})} \langle \varsigma_{x-L+1:1} | \varsigma_{x'-L+1:1} \rangle. \end{aligned} \quad (5.16)$$

Iteratively applying Eq. (5.16), we obtain

$$\langle \varsigma_{x-L:0} | \varsigma_{x'-L:0} \rangle = \sum_{x_{0:L}} \sqrt{\tilde{P}(x_{0:L} | x_{-L:0}) \tilde{P}(x_{0:L} | x'_{-L:0})}. \quad (5.17)$$

As the inferred probability distributions are imperfect distributions, we can use the same approach as Eq. (4.17) to work out how a perturbation affects exact probabilities to give the imperfect probabilities. We let $\tilde{P} = P + \Delta P$ where P is the exact distribution, and ΔP is some relative perturbation to the exact distribution. The square-root of two perturbed distributions $\tilde{P}$ and $\tilde{Q}$ can be simplified as,

$$\begin{aligned}
 \sqrt{\tilde{P}\tilde{Q}} &= \sqrt{(P + \Delta P)(Q + \Delta Q)} \\
 &= \sqrt{PQ + P\Delta Q + Q\Delta P + \Delta P\Delta Q} \\
 &= \sqrt{PQ} \left[1 + \frac{1}{2} \left(\frac{\Delta P}{P} + \frac{\Delta Q}{Q} + \frac{\Delta P\Delta Q}{PQ} \right) + \dots \right] \\
 &\approx \sqrt{PQ} + \frac{\sqrt{PQ}}{2} \left[\frac{\Delta P}{P} + \frac{\Delta Q}{Q} \right]
 \end{aligned} \tag{5.18}$$

to first order of Δ . Substituting $\tilde{P}(x_{0:L}|x_{-L:0})$ and $\tilde{P}(x_{0:L}|x'_{-L:0})$ in place of $\tilde{P}$ and $\tilde{Q}$ respectively, we have

$$\begin{aligned}
 \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle &\approx \sum_{x_{0:L}} \sqrt{P(x_{0:L}|x_{-L:0})P(x_{0:L}|x'_{-L:0})} + \\
 &\quad \frac{\sqrt{P(x_{0:L}|x_{-L:0})P(x_{0:L}|x'_{-L:0})}}{2} \left[\frac{\Delta P(x_{0:L}|x_{-L:0})}{P(x_{0:L}|x_{-L:0})} + \frac{\Delta P(x_{0:L}|x'_{-L:0})}{P(x_{0:L}|x'_{-L:0})} \right] \\
 &= \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle_{\text{exact}} + \Delta \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle.
 \end{aligned} \tag{5.19}$$

This results in the error

$$\begin{aligned}
 \Delta \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle &= \\
 \sum_{x_{0:L}} \frac{\sqrt{P(x_{0:L}|x_{-L:0})P(x_{0:L}|x'_{-L:0})}}{2} \left[\frac{\Delta P(x_{0:L}|x_{-L:0})}{P(x_{0:L}|x_{-L:0})} + \frac{\Delta P(x_{0:L}|x'_{-L:0})}{P(x_{0:L}|x'_{-L:0})} \right].
 \end{aligned} \tag{5.20}$$

For a randomly selected L -length word $x_{-L:0}$, it is either $x_{-L:0}$ or not. Thus, $P(x_{0:L}|x_{-L:0})$ is said to follow a binomial distribution that can be approximated to a normal distribution if the stochastic process is of large length N . The standard

deviation of $P(x_{0:L}|x_{-L:0})$ is hence

$$\sigma_{P(x_{0:L}|x_{-L:0})} = \sqrt{P(x_{0:L}|x_{-L:0})(1 - P(x_{0:L}|x_{-L:0}))}. \quad (5.21)$$

We use the standard error of the mean to estimate the error of the mean value of $P(x_{0:L}|x_{-L:0})$. This gives $\sigma_{\bar{P}(x_{0:L}|x_{-L:0})} = \sqrt{\frac{P(x_{0:L}|x_{-L:0})(1 - P(x_{0:L}|x_{-L:0}))}{N}}$.

The magnitude of error $|\Delta P(x_{0:L}|x_{-L:0})|$ can be approximated as $|\sigma_{\bar{P}(x_{0:L}|x_{-L:0})}|$ to relate the existence of $\Delta P(x_{0:L}|x_{-L:0})$ to finite N ,

$$\begin{aligned} |\Delta P(x_{0:L}|x_{-L:0})| &\approx |\sigma_{\bar{P}(x_{0:L}|x_{-L:0})}| \\ &\approx \sqrt{\frac{P(x_{0:L}|x_{-L:0})(1 - P(x_{0:L}|x_{-L:0}))}{N}}. \end{aligned} \quad (5.22)$$

The conditional probabilities $\sum_{x_{0:L}} P(x_{0:L}|x_{-L:0}) = 1$. Assuming that all the conditional probabilities have equal probability of occurrence, we can use the approximation that $P(x_{0:L}|x_{-L:0}) \approx \frac{1}{|A|^L}$. By keeping that L fixed, we observe that $|\Delta \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle|$ scales proportionally to $\frac{1}{\sqrt{N}}$. This means that the error to the fidelity $|\Delta \langle s_{x_{-L:0}} | s_{x'_{-L:0}} \rangle|$ is directly related to length of the stochastic process, N . In this work, we set the measure of statistical similarity to have a δ -tolerance of $\delta = \frac{1}{2\sqrt{N}}$. The factor of 2 gives a more stringent criteria for merging quantum memory states.

Figure 5.2: (a) Unitary quantum models operate with a memory qubit $|\sigma_j\rangle$ and an ancillary blank qubit $|0\rangle$ that will undergo unitary operation to produce $|\sigma_{\lambda(x,j)}\rangle$ and $|x\rangle$. (b) In the quantum circuit picture, q_0 functions as the memory (control) qubit that acts on target ancillary blank qubits q_1 , q_2 , q_3 , and so on. (c) Each unitary in (b) is decomposed in a series of rotations, CNOTs, NOT gates, and the multiplexor gate.

CHAPTER 6

Quantum Error Mitigation

Feeling insignificant because the universe is large has exactly the same logic as feeling inadequate for not being a cow.

— David Deutsch

Armed with the Quantum Inference Protocol in Chapter [4](#) and the unitary construction and decomposition in Chapter [5](#), we now have the necessary tools to simulate the inferred phaseless unitary quantum models of stochastic processes on a quantum computer. Thus far, previous work has focused mainly on the aspect of searching for and improving existing quantum causal models with the lowest memory requirements. Quantum causal models have also been experimentally demonstrated in proof-of-principle spatially tailored optical quantum processors [\[72–74\]](#). With quantum computing gaining ever increasing attention alongside the availability of open source cloud quantum computing [\[75–78\]](#), there is greater opportunity to simulate quantum causal models on quantum computers.

However, challenges remain. The existing platforms for quantum computing are Noisy Intermediate-Scale Quantum (NISQ) devices [26], implying that any measurement outcome will be heavily influenced by noise. Traditionally, error correction codes are used to suppress noise for reliability [29–32], but typically demand large number of qubits, sacrificing feasibility and practicality for accuracy. Error mitigation techniques were introduced [33–40] as an alternate method to quantum error correction for noise suppression. Error mitigation techniques typically require far less qubits than error correction codes with more focus on post-processing measurement results on classical computers, thereby making them more practical and appealing.

In this Chapter, we aim to implement a two-qubit phaseless unitary quantum model enhanced with the error mitigation method known as the probabilistic error cancellation method [33, 37, 79] to reduce the effects of noise [41]. The platform of choice is the IBM Quantum Experience. We conduct the simulation on local hardware due to the immense computational resource demand for a system of two qubits on the real quantum backend. As such, a noise model is first extracted from *ibmq_toronto*, one of the IBM Quantum Falcon processors, before performing the simulation on local hardware. Our numerical results demonstrates success in achieving more accurate results than without error mitigation, which can be easily quantified with any distance measure such as fidelity. We also investigate the use of limited number of shots in gate set tomography (which makes up the pre-experimental portion) and its effect on eventual simulation accuracy. This work serves as the ideal stepping stone for future quantum modelling obtained with error mitigation on NISQ devices.

6.1 Methodology for error mitigation

The workflow for the probabilistic error cancellation method consists of four key steps illustrated in Fig. 6.1. The Pauli transfer matrix notation is used for the probabilistic

Figure 6.1: The general workflow for the error mitigation method comprises of 4 main steps.

error cancellation method and is covered in Chapter [2.2.5](#).

6.1.1 Gate set tomography.

Gate set tomography [\[80\]](#) is performed to obtain the statistics of noise for state preparations, measurement outcomes, and operations. It comprises of three steps.

- (1) Initialise a circuit with one of four linearly independent initial states

$$\bar{\rho}_k \in \{ |0\rangle\langle 0|, |1\rangle\langle 1|, |+\rangle\langle +|, |y+\rangle\langle y+| \}. \quad (6.1)$$

$|+\rangle \equiv H|0\rangle$ is the eigenstate of the Pauli operator σ_x and $|y+\rangle \equiv SH|0\rangle$ is the eigenstate of the Pauli operator σ_y , both with eigenvalue 1. H is the Hadamard gate while S is a $\sqrt{Z}$ gate. Let $\bar{\rho}_{k=1} = |0\rangle\langle 0|$, $\bar{\rho}_{k=2} = |1\rangle\langle 1|$, $\bar{\rho}_{k=3} = |+\rangle\langle +|$, and $\bar{\rho}_{k=4} = |y+\rangle\langle y+|$.

- (2) Insert the operator(s) $\bar{O}^{(l)}$ that make up the circuit. We label each operator with superscript (l) . If there is only one operator, then the superscripts may be dropped. Replace $\bar{O}^{(l)}$ with $\bar{B}_i$ when necessary. Here, $\bar{O}^{(l)}$ are operators that make up the circuit and $\bar{B}_i$ are the basis operations used to encode the inverse noise.
- (3) Measure the expectation values for four linearly independent observables of the Pauli operators $\bar{M}_j \in \{I, \sigma_x, \sigma_y, \sigma_z\}$ for each quantum circuit. Specifically

this means measuring the expectation values for $\langle \bar{M}_{j=1} \rangle = \langle I \rangle$, $\langle \bar{M}_{j=2} \rangle = \langle \sigma_x \rangle$, $\langle \bar{M}_{j=3} \rangle = \langle \sigma_y \rangle$, and $\langle \bar{M}_{j=4} \rangle = \langle \sigma_z \rangle$.

A system of n qubits demands 4^n initial states requires

$$\bar{\rho}_k^{\otimes n} \in \{ |0\rangle\langle 0|, |1\rangle\langle 1|, |+\rangle\langle +|, |y+\rangle\langle y+| \}^{\otimes n} \quad (6.2)$$

and 4^n measurement bases

$$\bar{M}_j^{\otimes n} \in \{ I, \sigma_x, \sigma_y, \sigma_z \}^{\otimes n}. \quad (6.3)$$

6.1.2 Quasiprobability decomposition.

The noise profile for the initial states and measurement bases (without operators) can be obtained when step (2) is omitted. This "empty" circuit gives a 2^n by 2^n Gram matrix g with elements

$$g_{j,k} = \text{tr}(\bar{M}_j \bar{\rho}_k). \quad (6.4)$$

The noise profile for the operators can be found by first computing $\tilde{\mathcal{O}}^{(l)}$ that has elements

$$\tilde{\mathcal{O}}_{j,k}^{(l)} = \text{tr}(\bar{M}_j \tilde{\mathcal{O}}^{(l)} \bar{\rho}_k) \quad (6.5)$$

before computing

$$\hat{\mathcal{O}}^{(l)} = A g^{-1} \tilde{\mathcal{O}}^{(l)} A^{-1} \quad (6.6)$$

using an invertible matrix $A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \end{bmatrix}$ that are the Pauli operators basis sorted as $I, \sigma_x, \sigma_y, \sigma_z$ which the authors of Ref. [37] recommend using to assume error-free state preparation. Finally, the inverse noise $(\mathcal{N}^{(l)})^{-1}$ of each operator can be

calculated,

$$(\mathcal{N}^{(l)})^{-1} = \mathcal{O}^{(l),\text{exact}} (\hat{\mathcal{O}}^{(l)})^{-1}. \quad (6.7)$$

$\mathcal{O}^{(l),\text{exact}}$ is the exact Pauli transfer matrix representation of the operator(s) which can be found using Eq. (2.14). For small amounts of noise, $(\mathcal{N}^{(l)})^{-1} \approx I$.

Knowledge of the inverse noise $(\mathcal{N}^{(l)})^{-1}$ is crucial, as $\mathcal{N}^{(l)}$ indicates the noise $\bar{\mathcal{O}}^{(l)}$ incurs. With the inverse noise, one can now find ways to implement $(\mathcal{N}^{(l)})^{-1}$ after every $\bar{\mathcal{O}}^{(l)}$, thereby effectively removing the noise from $\bar{\mathcal{O}}^{(l)}$. In the probabilistic error cancellation method, $(\mathcal{N}^{(l)})^{-1}$ is encoded into a set of basis operations $\{\hat{\mathcal{B}}_i\}$. Each $\hat{\mathcal{B}}_i$ is then applied after $\bar{\mathcal{O}}^{(l)}$, essentially removing the noise. Each $\hat{\mathcal{B}}_i$ is obtained in a similar method by replacing operations $\bar{\mathcal{O}}^{(l)}$ with $\bar{\mathcal{B}}_i$ in the three steps and perform the same computation in Eq. (6.6).

We employ basis operations from Ref. [81] which consists of deterministic operations and is applicable for any noise model. There are 13 basis operations for a single qubit system, and $13^2 + 72 = 241$ basis operations for a system of two qubits. The $13^2 = 169$ basis operations are straightforwardly found with tensor products of the 13 basis operations while the remaining 72 basis operations are made up of CNOTs, controlled-phase, controlled-Hadamards, CNOTs with eigenstates of the Hadamard gate, SWAP, and iSWAP gates. The 241 basis operations are listed in Tables 6.1 and 6.2.

There are other sets of basis operations available such as the one recommended in Ref. [37] but their use requires post-selection. Post-selection increases the difficulty of implementing basis operations as there needs to be a conditional statement—even before post-processing of Monte Carlo results—that accepts the measurement results when some condition is met, or reject otherwise. One could choose a more general class of basis operations that includes a continuous set of noisy implementable operations [81–85], which could reduce the sampling cost characterised in Eq. (6.12).

This detailed investigation on extended basis operations is left for future work.

The inverse noise is encoded into a set of basis operations $\{\hat{\mathcal{B}}_i\}$ with a quasiprobability decomposition:

$$(\mathcal{N}^{(I)})^{-1} = \sum_i q_{\mathcal{O}^{(I)},i} \hat{\mathcal{B}}_i \quad (6.8)$$

Letting $|\hat{\rho}_k\rangle\rangle = A_{\bullet,k}$ and $\langle\langle \hat{Q}_j| = (gA^{-1})_{j,\bullet}$, where $M_{\bullet,k}$ is the k^{th} column of matrix M and $M_{j,\bullet}$ is the j^{th} row of matrix M . Quasiprobabilities of initial states and measurement bases are found as

$$|\rho^{\text{exact}}\rangle\rangle = \sum_k q_{\rho,k} |\hat{\rho}_k\rangle\rangle \quad (6.9)$$

where $\rho^{\text{exact}} = |0\rangle\langle 0|$,

$$\langle\langle M^{\text{exact}}| = \sum_j q_{M,j} \langle\langle \hat{M}_j| \quad (6.10)$$

where M^{exact} is a Z -basis measurement.

Table 6.1: The 13 CPTP maps listed here are quantum channels that can be used for basis operations. id represents the identity channel while $\mathcal{P}_{|\cdot\rangle}$ signifies a state preparation channel that prepares the state $|\cdot\rangle$.

$\bar{\mathcal{B}}_1$	id	$\bar{\mathcal{B}}_8$	$\mathcal{H}\mathcal{S}\mathcal{S}^\dagger$
$\bar{\mathcal{B}}_2$	$\mathcal{X}$	$\bar{\mathcal{B}}_9$	$\mathcal{H}$
$\bar{\mathcal{B}}_3$	$\mathcal{Y}$	$\bar{\mathcal{B}}_{10}$	$\mathcal{H}^\dagger\mathcal{S}^\dagger\mathcal{H}\mathcal{S}\mathcal{H}$
$\bar{\mathcal{B}}_4$	$\mathcal{Z}$	$\bar{\mathcal{B}}_{11}$	$\mathcal{P}_{ +\rangle}$
$\bar{\mathcal{B}}_5$	$\mathcal{H}^\dagger\mathcal{S}^\dagger\mathcal{H}$	$\bar{\mathcal{B}}_{12}$	$\mathcal{P}_{ y+\rangle}$
$\bar{\mathcal{B}}_6$	$\mathcal{H}\mathcal{S}\mathcal{S}^\dagger\mathcal{H}^\dagger\mathcal{S}^\dagger$	$\bar{\mathcal{B}}_{13}$	$\mathcal{P}_{ 0\rangle}$
$\bar{\mathcal{B}}_7$	$\mathcal{S}^\dagger$		

Table 6.2: A system of $n = 2$ qubits requires 241 basis operations. The first 169 can be obtained by taking the tensor product of the 13 basis operations listed in Table 6.1 while the remaining 72 basis operations are listed here. Here, we let $\mathcal{K} = \mathcal{S} \circ \mathcal{H}$.

$\bar{\mathcal{B}}_{170\text{to}178}$	$\mathcal{C}\mathcal{X} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{179\text{to}187}$	$\mathcal{X}_1 \circ \mathcal{C}\mathcal{X} \circ \mathcal{X}_1 + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{188\text{to}196}$	$\mathcal{C}\mathcal{S} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{197\text{to}205}$	$\mathcal{C}\mathcal{H} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{206\text{to}178}$	$\mathcal{C}_{\mathcal{H}}\mathcal{X} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{215\text{to}223}$	$\mathcal{C}\mathcal{X} \circ \mathcal{H}_1 + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$
$\bar{\mathcal{B}}_{224\text{to}226}$	$\mathcal{S}_{\text{swap}} + \text{conjugation with } \mathcal{K}_2, \mathcal{K}_2^\dagger$
$\bar{\mathcal{B}}_{227\text{to}232}$	$i\mathcal{S}_{\text{swap}} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_2^\dagger$
$\bar{\mathcal{B}}_{233\text{to}241}$	$\mathcal{S}_{\text{swap}} \circ \mathcal{H}_1 + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$

As a follow up to Table 6.2, " $\mathcal{U} + \text{conjugation with } \mathcal{K}_{1,2}, \mathcal{K}_{1,2}^\dagger$ " means:

$$\begin{aligned}
& \mathcal{K}_1 \otimes \mathcal{K}_2 \circ \mathcal{U} \circ \mathcal{K}_1^\dagger \otimes \mathcal{K}_2^\dagger \\
& \mathcal{K}_1 \otimes \mathcal{K}_2^\dagger \circ \mathcal{U} \circ \mathcal{K}_1^\dagger \otimes \mathcal{K}_2 \\
& \mathcal{K}_1 \otimes id \circ \mathcal{U} \circ \mathcal{K}_1^\dagger \otimes id \\
& \mathcal{K}_1^\dagger \otimes \mathcal{K}_2 \circ \mathcal{U} \circ \mathcal{K}_1 \otimes \mathcal{K}_2^\dagger \\
& \mathcal{K}_1^\dagger \otimes \mathcal{K}_2^\dagger \circ \mathcal{U} \circ \mathcal{K}_1 \otimes \mathcal{K}_2 \\
& \mathcal{K}_1^\dagger \otimes id \circ \mathcal{U} \circ \mathcal{K}_1 \otimes id \\
& id \otimes \mathcal{K}_2 \circ \mathcal{U} \circ id \otimes \mathcal{K}_2^\dagger \\
& id \otimes \mathcal{K}_2^\dagger \circ \mathcal{U} \circ id \otimes \mathcal{K}_2 \\
& id \otimes id \circ \mathcal{U} \circ id \otimes id
\end{aligned} \tag{6.11}$$

6.1.3 Monte Carlo simulation.

For implementation using a Monte Carlo simulation, we compute the cumulative distribution functions of $q_{\mathcal{O}^{(i)},j}$, $q_{p,k}$, and $q_{M,j}$, also finding the following values in the

Figure 6.2: Initialise each Monte Carlo run with state $\bar{\rho}_k = |q_0 q_1\rangle \langle q_0 q_1|$, apply operation $\bar{\mathcal{O}}^{(l)}$ (in this case it is the unitary operator U), apply basis operation $\hat{B}_i$, and measure in basis $\bar{M}_j$. Let the measured outcome be μ .

process which are indicative of the sampling cost,

$$C_{\mathcal{O}^{(l)}} = \sum_i |q_{\mathcal{O}^{(l)},i}|; \quad C_\rho = \sum_k |q_{\rho,k}|; \quad C_M = \sum_j |q_{M,j}|. \quad (6.12)$$

Given these probabilities

$$P(q_{\rho,k}) = \frac{|q_{\rho,k}|}{C_\rho}; \quad P(q_{B_i}) = \frac{|q_{\mathcal{O}^{(l)},i}|}{C_{\mathcal{O}^{(l)},i}}; \quad P(q_M) = \frac{|q_{M,j}|}{C_M}, \quad (6.13)$$

initialise one instance of a quantum circuit for one Monte Carlo run. The circuit is initialised in state $\bar{\rho}_k$ with some probability $P(q_{\rho,k})$, operation $\bar{\mathcal{O}}^{(l)}$ applied, some basis operation $\hat{B}_i$ chosen with probability $P(q_{B_i})$, then measured in basis $\bar{M}_j$ with probability $P(q_M)$. Each Monte Carlo run gives a measurement outcome $x \in \{0, 1\}$.

In our simulation, we replace $\bar{\mathcal{O}}^{(l)}$ with the unitary operator for the unitary quantum models and the circuit is illustrated in Fig. [6.2](#).

6.1.4 Post-processing results.

With N_{MC} Monte Carlo runs giving N_{MC} measurement outcomes, one first assigns a sign to each measurement outcome associated with each Monte Carlo circuit. The

sign is found by

$$\begin{aligned} \text{sgn} &= \text{sign} \left(q_{\rho,k} \left[\prod_l q_{\mathcal{O}(l)} \right] q_{M,j} \right) \\ &= \text{sign} (q_{\rho,k} [q_{\mathcal{O}(1)} q_{\mathcal{O}(2)} \dots q_{\mathcal{O}(l)}] q_{M,j}). \end{aligned} \quad (6.14)$$

Then, one can compute the estimated error mitigated expectation values as

$$\begin{aligned} P_{\text{QEM}}(x = 0) &= C \left(\frac{\text{No. of } x = 0 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x = 0 | \text{sgn} = -1}{N_{\text{MC}}} \right) \\ P_{\text{QEM}}(x = 1) &= C \left(\frac{\text{No. of } x = 1 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x = 1 | \text{sgn} = -1}{N_{\text{MC}}} \right) \end{aligned} \quad (6.15)$$

where

$$\begin{aligned} C &= [C_{\rho} \prod_l C_{\mathcal{O}(l)} C_M] \\ &= C_{\rho} [C_{\mathcal{O}(1)} C_{\mathcal{O}(2)} \dots C_{\mathcal{O}(l)}] C_M. \end{aligned} \quad (6.16)$$

The same methodology can be extended to multi-steps. For instance, the 2-step estimated error mitigated expectation values are given by

$$\begin{aligned} P_{\text{QEM}}(x^{(t=1)} = 0, x^{(t=2)} = 0) &= C \left[\frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 00 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 00 | \text{sgn} = -1}{N_{\text{MC}}} \right] \\ P_{\text{QEM}}(x^{(t=1)} = 0, x^{(t=2)} = 1) &= C \left[\frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 01 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 01 | \text{sgn} = -1}{N_{\text{MC}}} \right] \\ P_{\text{QEM}}(x^{(t=1)} = 1, x^{(t=2)} = 0) &= C \left[\frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 10 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 10 | \text{sgn} = -1}{N_{\text{MC}}} \right] \\ P_{\text{QEM}}(x^{(t=1)} = 1, x^{(t=2)} = 1) &= C \left[\frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 11 | \text{sgn} = +1}{N_{\text{MC}}} - \frac{\text{No. of } x^{(t=1)} x^{(t=2)} = 11 | \text{sgn} = -1}{N_{\text{MC}}} \right]. \end{aligned} \quad (6.17)$$

Superscripts (t) indicate the index of the multi-step distribution. With Eq. (6.15) and (6.17), one can easily extrapolate to find further multi-step error mitigated

distributions ($t \geq 3$). Expectation values from the Monte Carlo simulation incurs a standard deviation of $\sigma_{MC} = \frac{C}{\sqrt{N_{MC}}}$. Hence, the Monte Carlo simulation is repeated many times until the standard deviation $\sigma_{MC} = \frac{C}{\sqrt{N_{MC}}}$ is within an acceptable value.

6.2 Results

Figure 6.3: The perturbed coin ϵ -machine is used as a generator to generate a stochastic process with parameter $p = 0.2$. After using the Quantum Inference Protocol to generate quantum memory states, the unitary operator can be constructed and decomposed. The statistical distribution of outputs can then be obtained with and without error mitigation to compare the efficacy of error mitigation.

We are now equipped with the tools to simulate a stochastic process with the inferred quantum causal model on a quantum device and carry out predictive capabilities. Noise from NISQ devices tend to skew the output distribution of the futures away from the exact distribution. As such, we enhance the inferred quantum causal model with the probabilistic error cancellation method as part of error mitigation. This will enable the inferred quantum causal model to produce an output distribution that is closer to the exact distribution.

We demonstrate this with the perturbed coin process as its unifilar and Markovian nature greatly simplifies the methodology; phaseless quantum memory states can be constructed with $L \geq 1$ accompanied with a decent-length stochastic process such as $N \geq O(10^4)$. The workflow is depicted in Fig. 6.3. First, the ϵ -machine for the perturbed coin is tuned some value of p ($0 < p < 1$, $p \neq 0.5$) and a single shot of stochastic process of length $N \geq O(10^4)$ is generated. Next, the Quantum Inference

Protocol takes the stochastic process as the input. With $L \geq 1$, it generates a set of phaseless quantum memory states which undergo merging if states their fidelity is within a δ -tolerance with $\delta = \frac{1}{2\sqrt{N}}$, see Eq. (5.1). The merged quantum memory states allow the unitary operator to be constructed and decomposed into elementary quantum gates. Then, gate set tomography is performed for 16 initial states, 16 measurement bases, the unitary operator, and 241 basis operations. The inverse noise of the unitary is obtained and the Monte Carlo method is carried out to obtain N_{MC} samples. Lastly, the Monte Carlo results are post-processed to give the error mitigated estimated expectation values of the outcomes. The error mitigated results will be compared against the non-error mitigated (noisy) results.

We stress again that all results are based on simulation performed on local hardware due to the immense computational resource demand for a system of two qubits on the actual quantum device. The noise model used for our simulation is from *ibmq_toronto*.

6.2.1 Multi-step error mitigation

Multi-step output results from the Monte Carlo simulation are post-processed to form a statistical distribution. The statistical distribution will approach a valid probability distribution with all elements in the distribution being real and positive, bounded between 0 and 1, as N_{MC} increases.

The results in Fig. 6.4 are valid probability distributions based on $N_{MC} = 10^7$ Monte Carlo runs, each run resulting in one measurement outcome. An identical number of samples ($N_{noisy} = 10^7$) for the noisy non-error mitigated quantum model was obtained on the simulator as well. From the plots in Fig. 6.4, it is evident that error mitigation improves the statistical distribution of the outcomes for both 1 and 2 time steps of unitary operation. If the statistical distribution forms a valid probability

Figure 6.4: The multi-step joint statistical distributions for (a) 1 time step of unitary operation with outputs given by $x_{\text{output}} \equiv x^{(t=1)} \in \{0, 1\}$ and (b) 2 time steps of unitary operator with outputs given by $x_{\text{outputs}} \equiv x^{(t=1)}x^{(t=2)} \in \{00, 01, 10, 11\}$, all of which are obtained with $N_{\text{MC}} = 10^7$ samples.

distribution, one has the freedom to choose any choice of probability distance measure to quantify the closeness of two distributions. We use the fidelity measure in Fig. 6.4 to exemplify the effectiveness of error mitigation method.

The Monte Carlo sampling produces a normal distribution for each outcome (i.e. $x_{(t=1)} \in \{0, 1\}$, $x_{(t=2)} \in \{00, 01, 10, 11\}$, and larger t). These distributions have standard deviation $\sigma_{\text{MC}} = \frac{C}{\sqrt{N_{\text{MC}}}}$ where C is given in Eq. (6.16) and indicates the sampling cost. For multi-step unitary operation, the standard deviation is compounded

each time the unitary operation is carried out.

$$\sigma_{\text{MC}}^{(t)} = \frac{C^t}{\sqrt{N_{\text{MC}}}}. \quad (6.18)$$

In the example, $C \approx 20$ and the corresponding standard deviation for a single time step output $x^{(t=1)}$ is $\sigma_{\text{MC}}^{(t=1)} \approx 20/\sqrt{10^7} = 0.0063$, two time steps outputs $x^{(t=1)}x^{(t=2)}$ is $\sigma_{\text{MC}}^{(t=2)} \approx (20)^2/\sqrt{10^7} = 0.1265$, and three time steps outputs $x^{(t=1)}x^{(t=2)}x^{(t=3)}$ is $\sigma_{\text{MC}}^{(t=3)} \approx (20)^3/\sqrt{10^7} = 2.5298$. This simple calculation implies that 3 time steps of unitary operation is highly likely to produce a statistical distribution that is an invalid probability distribution.

Tracing back the origins of the factor C , it is observed that C is highly dependent on the amount of noise in the quantum device and will not be within any immediate control. As quantum technologies and hardware improves with time, C is expected to decrease, allowing for a more feasible error mitigation for multi-step outputs with less N_{MC} .

In fact, the factor C is also dependent on the number of shots used in gate set tomography (GST), denoted as N_{GST} . Should N_{GST} be small, the inverse noise $(\mathcal{N}^{(l)})^{-1}$ incur errors that trickle down to the quasiprobability decomposition of basis operations. These basis operations are also not spared from any errors arising from small N_{GST} . As such, these errors will compound and affect C . We derive an approximate scaling for errors in $(\mathcal{N}^{(l)})^{-1}$ and C with respect to N_{GST} in Section [6.2.4](#). The N_{GST} used in our example is $N_{\text{GST}} = 8192$ and it follows the maximal limit of the IBM Quantum backend *ibmq_toronto* to allow for more realistic simulation.

In addition, since the fidelity was used to quantify the distances between the error mitigated distribution and noisy distribution, we show how this scales with C and N_{MC} in Section [6.2.5](#).

6.2.2 Individual-step error mitigation

Post-processing the Monte Carlo results for individual-step error mitigation is possible as well, as illustrated in Fig. 6.5. Depending on the size of C , a smaller sample size would be sufficient to obtain decent statistics on the individual outputs as the standard deviation is not compounded for individual-step outputs. With $C \approx 20$, $N_{\text{MC}} \geq 10^6$ is required to have $\sigma_{\text{MC}} \leq 0.02$.

However, the disadvantage of post-processing individual-step outputs is that multi-step joint distributions would be close to impossible to reconstruct. This is because any correlation between individual outputs are lost when these individual outputs are processed. The multi-step joint distribution cannot be recovered unless the input stochastic process is a fully random process – only then will the following relation hold,

$$P(x_{0:K}) = P(x_0)P(x_1)P(x_2)\dots P(x_{K-1}), \quad (6.19)$$

where $x_{0:K}$ is the future outputs from K unitary operations.

Nonetheless, with $N_{\text{MC}} = 10^7$ at hand, we split the full $N_{\text{MC}} = 10^7$ dataset into 10^3 chunks with each chunk having $N_{\text{MC,chunk}} = 10^4$ samples. Splitting the full dataset into chunks simulates running Monte Carlo simulations in chunks and analysing each chunk individually. For each chunk, the individual error mitigated expectation value $P_{\text{QEM}}(x = 1)$ is computed before the collection of expectation values for $P_{\text{QEM}}(x = 1)$'s is plotted in a histogram. The histogram depicts the frequency of occurrence of each chunk's $P_{\text{QEM}}(x = 1)$. The standard deviation of the histogram is expected to be $\sigma_{\text{MC,chunk}} \approx \frac{20}{\sqrt{10^4}} = 0.2$.

In Fig. 6.5, the histograms illustrate how $P_{\text{QEM}}(x^{(t)} = 1)$ is distributed for up to $t = 3$. The mean is observed to be closer to the exact analytical value than the noisy value $P_{\text{noisy}}(x^{(t)} = 1)$, demonstrating the success of error mitigation method.

Figure 6.5: Distribution for the expectation values for $P_{\text{QEM}}(x^{(t)} = 1)$ i.e. for outcome $x = 1$ for (a) $t=1$ time step of unitary operation, (b) $t=2$ time steps of unitary operations, and (c) $t=3$ time steps of unitary operations. All figures had error mitigation carried out (blue histogram) and the results for the output 1 are compared against the ideal quantum simulator without noise (black) and quantum simulator with noise (green). The histograms are constructed with 10^3 chunks, each chunk having $N_{\text{MC,chunk}} = 10^4$ samples.

However, since $\sigma_{\text{MC,chunk}} \approx 0.2$, the spread is evidently large.

6.2.3 Memory costs

Thus far, we have seen that error mitigation is successful in reducing the effects of noise in Figs. 6.4 and 6.5. This begs the question: at what cost can they be considered successful? Entropic measures of memory carry the greatest operational meaning in the context of simultaneous simulation. This represents the average memory needed to simulate each instance of a stochastic process multiple times to build a statistical distribution of futures. In this context, we define the *cumulative memory cost* as the total memory that an arbitrary causal model requires to build a statistical distribution of futures using N_{model} samples where $N_{\text{model}} \gg 1$.

The ε -machine for the perturbed coin process has $C_\mu = 1$ at $0 \leq p \leq 1$ and $C_\mu = 0$ at $p = 0.5$. On the other hand, unitary quantum models have quantum statistical memory C_q that is equal to that in Eq. 3.18. It is continuous in the range $0 \leq p \leq 1$. For convenience,

$$C_q = - \left(\frac{1}{2} + \sqrt{p(1-p)} \right) \log_2 \left(\frac{1}{2} + \sqrt{p(1-p)} \right) - \left(\frac{1}{2} - \sqrt{p(1-p)} \right) \log_2 \left(\frac{1}{2} - \sqrt{p(1-p)} \right).$$

For instance at $p = 0.2$, the ε -machine for the perturbed coin process requires $C_\mu = 1$ bit of memory while unitary quantum models require $C_q = 0.4690$ bits of memory.

In a perfect noiseless scenario, the cumulative memory cost of unitary quantum models would be less than that of the ε -machine. Setting the standard deviation of the output statistical distribution to be $\sigma = 0.02$, $N_{\varepsilon\text{-machine}} = 2500$ samples are needed, and the cumulative memory cost is 2500 bits. Analogously, $N_{\text{q-models}} = 2500$ samples are needed for unitary quantum models as well, but its cumulative memory cost is $2500 \times 0.4690 = 1172.5$ bits. This illustrates the clear advantage in cumulative

Figure 6.6: The perturbed coin has the following C_μ and C_q for $0 \leq p \leq 1$. C_μ is discontinuous only at $p = 0.5$ as the ε -machine at $p = 0.5$ is essentially a random process that requires only a single causal state. The region where $0.4688 \leq p \leq 0.5134$ ($p \neq 0.5$) denotes the region where the cumulative memory cost for the quantum models require outdoes the memory cost for the ε -machine when error mitigation methods are taken into account.

memory that unitary quantum models have over ε -machines in the noiseless scenario.

However, simulating quantum models on NISQ devices requires error mitigation which incur some standard deviation σ_{MC} . For error mitigated quantum models to have a standard deviation of $\sigma_{MC} = 0.02$ as well, $N_{MC} = \left(\frac{C}{\sigma_{MC}}\right)^2 = \left(\frac{20}{0.02}\right)^2 = 10^6$ samples would be needed. The corresponding cumulative memory cost for error mitigated quantum models would be $10^6 \times 0.4690 = 469000$ bits, far exceeding that of the ε -machines, putting error mitigated quantum models at a clear disadvantage.

Observing that the p -value indirectly controls the degree of non-orthogonality between quantum memory states (which in turn impacts C_q), we ask, given the current climate of quantum computing on NISQ devices, how much non-orthogonality between quantum states is required before the cumulative memory cost of error mitigated models matches and outperforms the ε -machine? In other words, what p -values give rise to error mitigated quantum models having a cumulative memory

advantage over ϵ -machines? We solve the following inequality to find C_q .

$$\begin{aligned} N_{MC} \times C_q &\leq N_{\epsilon\text{-machine}} \times C_\mu \\ 10^6 \times C_q &\leq 2500 \times C_\mu \\ C_q &\leq 0.0025 \text{ bits.} \end{aligned} \tag{6.20}$$

The corresponding p -values for which this $C_q \leq 0.0025$ is $0.4866 \leq p \leq 0.5134$, $p \neq 0.5$. This region is demarcated with two vertical dotted lines in Fig. 6.6. We expect the range of p -values to increase as quantum hardware improves, decreasing the value of C and in turn reducing N_{MC} needed for the cumulative memory advantage.

6.2.4 Scaling for finite number of GST shots

Effect of finite N_{GST} -shots for gate set tomography. Finite N_{GST} shots for gate set tomography affects a few variables that contribute to the overall accuracy of the error mitigation. These variables will be written as a sum of the exact value and a fixed perturbation term Δ with scaling parameter ϵ to denote the strength of the perturbation.

- (1) Gram matrix g : Let $g_{N_{GST}}$ be the Gram matrix with N_{GST} shots. Thus, $g_{N_{GST}} = g + \epsilon\Delta g$.
- (2) $\tilde{O}^{(l)}$: Let $\tilde{O}_{N_{GST}}^{(l)} = \tilde{O}^{(l)} + \epsilon\Delta\tilde{O}^{(l)}$.
- (3) Basis operations $\tilde{B}_i$. Let $\tilde{B}_{i_{N_{GST}}} = \tilde{B}_i + \epsilon\Delta\tilde{B}_i^{(l)}$.

Our goal is to provide intuition of how the effects of finite N_{GST} shots in the pre-experimental portion cascade down to the cost of error mitigation. We begin with

$$\hat{O}^{(l)} = Ag^{-1}\tilde{O}^{(l)}A^{-1}. \tag{6.21}$$

For $\hat{\mathcal{O}}^{(l)}$ being affected by N_{GST} , we have

$$\begin{aligned}
 \hat{\mathcal{O}}_{N_{\text{GST}}}^{(l)} &= A g_{N_{\text{GST}}}^{-1} \tilde{\mathcal{O}}_{N_{\text{GST}}}^{(l)} A^{-1} \\
 &= A (g + \epsilon \Delta g)^{-1} (\tilde{\mathcal{O}}^{(l)} + \epsilon \Delta \tilde{\mathcal{O}}^{(l)}) A^{-1} \\
 &= A (g^{-1} - g^{-1} \epsilon \Delta g g^{-1}) (\tilde{\mathcal{O}}^{(l)} + \epsilon \Delta \tilde{\mathcal{O}}^{(l)}) A^{-1} \\
 &= A g^{-1} \tilde{\mathcal{O}}^{(l)} A^{-1} + A g^{-1} \epsilon \Delta \tilde{\mathcal{O}}^{(l)} A^{-1} - A g^{-1} \epsilon \Delta g g^{-1} \tilde{\mathcal{O}}^{(l)} A^{-1} + O(\epsilon^2).
 \end{aligned} \tag{6.22}$$

This gives

$$\begin{aligned}
 \hat{\mathcal{O}}^{(l)} &= A g^{-1} \tilde{\mathcal{O}}^{(l)} A^{-1} \\
 \epsilon \Delta \hat{\mathcal{O}}^{(l)} &= \epsilon [A g^{-1} \Delta \tilde{\mathcal{O}}^{(l)} A^{-1} - A g^{-1} \Delta g g^{-1} \tilde{\mathcal{O}}^{(l)} A^{-1}] + O(\epsilon^2).
 \end{aligned} \tag{6.23}$$

From this, we get $\|\epsilon \Delta \hat{\mathcal{O}}^{(l)}\| \sim \epsilon (\|\Delta \tilde{\mathcal{O}}^{(l)}\| + \|\Delta g\|)$ up to the first order of ϵ , where $\|\cdot\|$ is the operator norm. The error to $\epsilon \Delta \hat{\mathcal{O}}^{(l)}$ has contributions from $\epsilon \Delta \tilde{\mathcal{O}}^{(l)}$ and $\epsilon \Delta g$. More concretely, since the strength of perturbation is largely dominated by how many N_{GST} shots are performed, we set $\epsilon = 1$ and let Δ represent the full extent of perturbation due to N_{GST} shots. The error for basis operations $\hat{\mathcal{B}}_i$ can similarly be found by replacing the operators $\tilde{\mathcal{O}}^{(l)}$ in Eq. (6.22).

The inverse of noise can be found as

$$(\mathcal{N}^{(l)})^{-1} = \mathcal{O}^{(l), \text{exact}} (\hat{\mathcal{O}}^{(l)})^{-1}, \tag{6.24}$$

which means that for N_{GST} , the inverse noise $(\mathcal{N}_{N_{\text{GST}}}^{(l)})^{-1}$ is

$$\begin{aligned}
 (\mathcal{N}_{N_{\text{GST}}}^{(l)})^{-1} &= \mathcal{O}^{(l), \text{exact}} (\hat{\mathcal{O}}_{N_{\text{GST}}}^{(l)})^{-1} \\
 &= \mathcal{O}^{(l), \text{exact}} ((\hat{\mathcal{O}}^{(l)} + \Delta (\hat{\mathcal{O}}^{(l)}))^{-1}) \\
 &\approx \mathcal{O}^{(l), \text{exact}} ((\hat{\mathcal{O}}^{(l)})^{-1} - (\hat{\mathcal{O}}^{(l)})^{-1} \Delta \hat{\mathcal{O}}^{(l)} (\hat{\mathcal{O}}^{(l)})^{-1}) \\
 &= \mathcal{O}^{(l), \text{exact}} (\hat{\mathcal{O}}^{(l)})^{-1} - \mathcal{O}^{(l), \text{exact}} (\hat{\mathcal{O}}^{(l)})^{-1} \Delta \hat{\mathcal{O}}^{(l)} (\hat{\mathcal{O}}^{(l)})^{-1}.
 \end{aligned} \tag{6.25}$$

This implies $\|\Delta(\mathcal{N}^{(l)})^{-1}\| \sim \|\Delta\tilde{\mathcal{O}}^{(l)}\|$, and together with (6.23), we get $\|\Delta(\mathcal{N}^{(l)})^{-1}\| \sim \|\Delta\tilde{\mathcal{O}}^{(l)}\| + \|\Delta g\|$.

Should one print out the values of $(\mathcal{N}^{(l)})^{-1}$ for different values of N_{GST} , one will notice that the elements of $(\mathcal{N}^{(l)})^{-1}$ will increase with smaller N_{GST} , especially for the off-diagonal elements. This shows the effects of having finite N_{GST} shots.

We expect the quasiprobabilities in the quasiprobability decomposition from N_{GST} to fluctuate as well. Hence, we let each quasiprobability with index i to be $q_{\mathcal{O}^{(l)}, N_{\text{GST}}, i} = q_{\mathcal{O}^{(l)}, i} + \Delta q_{\mathcal{O}^{(l)}, i}$. To first order of Δ ,

$$\begin{aligned} (\mathcal{N}_{\text{GST}}^{(l)})^{-1} &= \sum_i q_{\mathcal{O}^{(l)}, N_{\text{GST}}, i} \hat{\mathcal{B}}_{N_{\text{GST}}, i} \\ (\mathcal{N}^{(l)} + \Delta\mathcal{N}^{(l)})^{-1} &= \sum_i (q_{\mathcal{O}^{(l)}, i} + \Delta q_{\mathcal{O}^{(l)}, i}) (\hat{\mathcal{B}}_i + \Delta\hat{\mathcal{B}}_i) \quad (6.26) \\ ((\mathcal{N}^{(l)})^{-1} - (\mathcal{N}^{(l)})^{-1} \Delta\mathcal{N}^{(l)} (\mathcal{N}^{(l)})^{-1}) &\approx \sum_i q_{\mathcal{O}^{(l)}, i} \hat{\mathcal{B}}_i + q_{\mathcal{O}^{(l)}, i} \Delta\hat{\mathcal{B}}_i + \Delta q_{\mathcal{O}^{(l)}, i} \hat{\mathcal{B}}_i. \end{aligned}$$

The error from $\Delta\mathcal{N}^{(l)}$ contributes to $q_{\mathcal{O}^{(l)}, i}$ by a factor of $\Delta\hat{\mathcal{B}}_i$ but more importantly incurs a perturbation $\Delta q_{\mathcal{O}^{(l)}, i}$. The fluctuation to $C_{\mathcal{O}^{(l)}}$ is hence $\Delta C_{\mathcal{O}^{(l)}} = \sum_i |\Delta q_{\mathcal{O}^{(l)}, i}|$.

Scaling with respect to N_{GST} . We are now in a position to ascertain how each variable scales with respect to N_{GST} . The matrix g has elements $g_{j,k}$ that are comprised of measuring an initialised state ρ_k in basis Q_j which yields outcomes 0 or 1, allowing it to be represented by a Bernoulli distribution. The argument holds for $\tilde{\mathcal{O}}^{(l)}$ (and $\tilde{\mathcal{B}}_i$) as well, where its matrix has elements $\tilde{\mathcal{O}}_{j,k}^{(l)}$ are built from outcomes 0 or 1 with some probabilities p or $1 - p$. Each matrix element is approximated to a normal distribution when N_{GST} is large enough, allowing the standard deviation to be $\sigma_p = \sqrt{p(1-p)}$. Invoking the standard error of the mean is used to estimate the error for the expectation value, $\sigma_{\bar{p}} = \sqrt{\frac{p(1-p)}{N_{\text{GST}}}}$, $\|\Delta g\|$, $\|\Delta\tilde{\mathcal{O}}^{(l)}\|$, and $\|\Delta\tilde{\mathcal{B}}_i\|$ all scale as $\frac{1}{\sqrt{N_{\text{GST}}}}$. This implies that $\|\Delta(\hat{\mathcal{B}}_i)^{-1}\| \approx \|\Delta(\mathcal{N}^{(l)})^{-1}\| \approx \frac{2}{\sqrt{N_{\text{GST}}}}$. And hence, $\Delta q_{\mathcal{O}^{(l)}, i}$

scales roughly as $\frac{2}{\sqrt{N_{\text{GST}}}}$. The eventual error to $C_{\mathcal{O}^{(l)}}$ is $\Delta C_{\mathcal{O}^{(l)}} = \sum_i |\Delta q_{\mathcal{O}^{(l)},i}|$.

6.2.5 Scaling of fidelity

Figure 6.7: A simple illustration of how a small number of Monte Carlo samples is more likely to produce a distribution that crosses into ill-defined regions. The vertical axis here represents the frequency of obtaining error mitigated expectation value of x .

As the number of Monte Carlo runs N_{MC} is increased, σ_{MC} will decrease. The shape of the resultant distribution of error mitigated expectation values will asymptote towards a delta function as $N_{\text{MC}} \rightarrow \infty$. This lowers the chances of which the distribution crosses into ill-defined regions. (See Fig. 6.7.)

We can choose any quantifier to quantify the distance between the error mitigated distribution and the exact, such as the fidelity. How would such distance measure scale with respect to N_{MC} ?

Let P_{ex} be the exact distribution and P_{sim} be the distribution obtained through simulation. P_{sim} denotes the distribution of error mitigated expectation values of the simulation as well as the noisy simulation that has no error mitigation. The fidelity between two distributions P_{ex} and P_{sim} is

$$F(P_{\text{ex}}, P_{\text{sim}}) = \sum_i \sqrt{(p_{\text{ex},i})(p_{\text{sim},i})}. \quad (6.27)$$

Suppose P_{sim} can be seen as a distribution that has some perturbation to P_{ex} . Then we have the following equation to second order of ΔP_{ex} ,

$$\begin{aligned}
 F(P_{\text{ex}}, P_{\text{ex}} + \Delta P_{\text{ex}}) &= \sum_i \sqrt{(p_{\text{ex},i})(p_{\text{ex},i} + \Delta p_{\text{ex},i})} \\
 &= \sum_i p_{\text{ex},i} \sqrt{1 + \frac{\Delta p_{\text{ex},i}}{p_{\text{ex},i}}} \\
 &\approx \sum_i p_{\text{ex},i} \left[1 + \frac{1}{2} \left(\frac{\Delta p_{\text{ex},i}}{p_{\text{ex},i}} \right) - \frac{1}{8} \left(\frac{\Delta p_{\text{ex},i}}{p_{\text{ex},i}} \right)^2 \right] \\
 &= \sum_i \left[p_{\text{ex},i} + \frac{1}{2} (\Delta p_{\text{ex},i}) - \frac{1}{8} \frac{(\Delta p_{\text{ex},i})^2}{p_{\text{ex},i}} \right].
 \end{aligned} \tag{6.28}$$

The conservation of probability dictates that $\sum_i p_{\text{ex},i} = p_{\text{sim},i} = 1$ it can be observed that $\sum_i \Delta p_{\text{ex},i} = 0$. Because fidelity is bounded by $0 \leq F \leq 1$ [86]. Therefore, Eq. (6.28) resolves to

$$\begin{aligned}
 F(P_{\text{ex}}, P_{\text{ex}} + \Delta P_{\text{ex}}) &= \sum_i \left[p_{\text{ex},i} + \frac{1}{2} (\Delta p_{\text{ex},i}) - \frac{1}{8} \frac{(\Delta p_{\text{ex},i})^2}{p_{\text{ex},i}} \right] \\
 &= 1 + 0 - \sum_i \frac{1}{8} \frac{(\Delta p_{\text{ex},i})^2}{p_{\text{ex},i}} \\
 &= F(P_{\text{ex}}, P_{\text{ex}}) - \Delta F
 \end{aligned} \tag{6.29}$$

where $\Delta F = \sum_i \frac{1}{8} \frac{(\Delta p_{\text{ex},i})^2}{p_{\text{ex},i}}$. For each time step t of unitary operation, the error mitigation technique's expectation values for each $p_{\text{sim},i}$ occurs with a standard deviation $\sigma_{\text{MC}}^{(t)} = \frac{C^t}{\sqrt{N_{\text{MC}}}}$. As such, $\Delta p_i = \frac{C^t}{\sqrt{N_{\text{MC}}}}$. Substituting this, we obtain the scaling for ΔF as

$$\begin{aligned}
 \Delta F &= \sum_i \frac{1}{8 p_{\text{ex},i}} \left(\frac{C^t}{\sqrt{N_{\text{MC}}}} \right)^2 \\
 &= \frac{C^{2t}}{8 N_{\text{MC}}} \sum_i \frac{1}{p_{\text{ex},i}}.
 \end{aligned} \tag{6.30}$$

Therefore, the perturbation to the fidelity scales as $\sim \frac{C^{2t}}{N_{\text{MC}}}$.

Summary for quantum error mitigation

Error mitigation techniques are able to reduce the effects of noise. This is illustrated with the perturbed coin process after the unitary is constructed and decomposed into elementary quantum gates.

The error mitigation technique uses gate set tomography to obtain the inverse noise profile, then encodes the inverse noise profile into a set of basis operations. The Monte Carlo simulation is then used to sample and compute expectation values.

Multi-step error mitigation sees a compounded standard deviation

$$\sigma^t = \frac{C^t}{\sqrt{N_{MC}}}, \quad (6.31)$$

for each t time the unitary operator is used to generate the next $|x\rangle$ of the stochastic process.

Individual-step error mitigation has no compounded standard deviation, i.e.

$$\sigma = \frac{C}{\sqrt{N_{MC}}} \quad (6.32)$$

but any correlated information between the outputs is lost, rendering the reconstruction of multi-step distribution from individual-steps close to impossible.

CHAPTER 7

Classifying Elementary Cellular Automata

*First, think.
Second, believe.
Third, dream.
And finally, dare.*

- Walt Disney

In Chapter [4](#), we introduced the Quantum Inference Protocol and its purpose in the grand scheme of modelling stochastic processes with quantum mechanics directly through inference. We have also seen in Chapter [5](#) how to construct and decompose the appropriate unitary operator for a set of given quantum memory states, as well as how error mitigation can help to improve the fidelity of the expected outcomes when quantum causal models are simulated with real IBM Quantum Experience backend's noise model in Chapter [6](#). We briefly mentioned that the Quantum Inference Protocol has the capability to perform a model-independent estimate of the cost required for the quantum causal model to work, meaning it can infer $\tilde{C}_q$ from a stochastic process

without constructing the quantum memory states. This Chapter will demonstrate how model-independent inference of $\tilde{C}_q$ can be used to infer the amount of structure within elementary cellular automata (ECA) in the event where numerical simulations are paramount to determine the behaviour of ECA. Our work sheds light on a new classification scheme for ECA that comprises of a hybrid between discrete and continuous.

7.1 Cellular Automata

A cellular automaton is a simple dynamical system that has the potential to display rich diversity between order and randomness. Each cell in cellular automata updates itself deterministically based on its nearest neighbour's states. There are various types of cellular automata with the most notable one being the Game of Life. The Game of Life is based on two dimensions of space and one dimension of time and has been shown to be capable of generating patterns that eventually lead to a universal computer [87, 88]. Despite the feature-rich capabilities, scaling down to one dimension of space poses its challenges in terms of understanding the behaviour of such cellular automata.

Elementary cellular automata (ECA) operates on of one dimension of space and one dimension of time. The state of each cell is denoted with a variable $y_i^{(t)}$ where t represents the time index and i represents the spatial index. y is given by a set of symbols $\mathcal{A}$. In theory, $\mathcal{A}$ can contain an infinite number of alphabets and the state of the cell can be dependent on as many past states stretching arbitrarily far back in time.

In the case of elementary cellular automata, the automaton is made up of a one dimensional chain of cells with only two symbols, $\mathcal{A} \in \{\text{white, black}\} \equiv \{0, 1\} \equiv \{\text{dead, alive}\}$. Each cell is dependent on its immediate past and immediate past

Figure 7.1: (a) Each cell's state in elementary cellular automata (ECA) is updated by simple local update rules. These rules are permutations of its past nearest neighbours cells' states and its past state. A rule number is determined by converting the outcomes of states at t from decimal to binary, for example (a) shows ECA Rule $\{00011010\}_{\text{binary}} = \{26\}_{\text{decimal}}$. (b) Four qualitative classes of ECA are observed when the ECA is initialised with random initial conditions.

nearest neighbours. The state of a cell is thus a function of its immediate past and past nearest neighbours,

$$\mathcal{F} : y_i^{(t)} = \mathcal{F}(y_{i-1}^{(t-1)}, y_i^{(t-1)}, y_{i+1}^{(t-1)}), \quad (7.1)$$

yielding $2^{2^3} = 256$ different possible permutations called *rules* [89–92]. A rule number is determined by converting the string of possible outcomes from binary into decimal as illustrated in Fig. 7.1(a).

Symmetries exist among the 256 rules [93]. Mirror imaged symmetries have update functions

$$\mathcal{F}(y_{i+1}^{(t-1)}, y_i^{(t-1)}, y_{i-1}^{(t-1)}); \quad (7.2)$$

inverse symmetries (black on white) symmetries have update functions

$$\mathcal{F}(\bar{y}_{i-1}^{(t-1)}, \bar{y}_i^{(t-1)}, \bar{y}_{i+1}^{(t-1)}), \quad (7.3)$$

where the over-bar represents the inverse (where if $y_i = 1$, $\bar{y}_i = 0$); concurrent mirror imaged and inverse symmetries have update functions

$$\mathcal{F}(\bar{y}_{i-1}^{(t+1)}, \bar{y}_i^{(t-1)}, \bar{y}_{i+1}^{(t-1)}). \quad (7.4)$$

After accounting for rules that exhibit symmetries, 88 independent rules remain. The 88 independent rules evolves ECA (that are initialised with random initial conditions) to various types of behaviour that were first qualitatively classified into four classes [89, 90, 94] as depicted by Fig. 7.1(b).

Allowing an initial condition to evolve under different rules produces a rich diversity between order and randomness. Initial observations of how elementary cellular automata evolved with time led Wolfram to qualitatively classify the rules into four broad classes:

Class I All cells evolve and eventually arrive at the same state, making Class I rules trivial.

Class II The ECA shows periodic behaviour; the ECA repeats itself and may have spatial shifts.

Class III These are known as chaotic rules as the ECA is sensitive to initial conditions and appears to evolve in a random-like manner.

Class IV Complex structures are formed and are able to interact with each other, earning the title ‘complex rules’. These structures are termed as gliders.

7.2 Classifying Cellular Automata

Wolfram’s classification scheme was met with a myriad of attempts to classify ECA such as power spectra [95, 96], Langton parameter [93, 97–100], hierarchical

classification, topological classification [101–104], among others [105–111]. However, the boundaries between classes, especially that of Classes III and IV, still lack universal consensus.

For example, the power spectra of ECA [95] and mean-field clustering [93] agree that Wolfram’s Class III rules are very similar; the power spectrum analysis of Class III rules describe Class III as having broad band noise while the mean-field descriptions place Class III rules as globally chaotic. The place of disagreement is with Class IV rules. The power spectra analysis shows low frequencies for Class IV rules while the mean-field description of rules with glider systems (54 and 110) are classified as globally chaotic.

In another example, it can be easily shown that Rule 62 has extremely long transients before reaching periodicity, a Class II behaviour. Yet the power spectra of Rule 62 is similar to that of Class IV rules. In addition, while power spectra is used in both Ref. [95] and Ref. [96], Ref. [96] reveals that there are 59 spectral classes whereas Ref. [95] contains only 6 classes.

Also, Ref. [110] features a discrete classification scheme when initialised with a single black cell, and is able to group ECAs according to the patterns they generate: trivial, diagonal, fractals. or randomness. This classification scheme can identify differences between Rule 30 (pseudo-random number generator) and Rule 110 (capable of universal computation) but since the initial conditions are not i.i.d., the glider patterns in Rule 54 and kinks in Rule 18 are not visible.

To further emphasise the lack of consensus between classification schemes, Ref. [108] places rules 30, 22, 18, 122, 54, and 110 in the same class, grouping together pseudo random with universal computation capable, which may be counterintuitive of their relative complexities.

Given the classification dilemma, what would be an appropriate method of classifying the 88 independent rules? There is hardly any dispute between Wolfram’s Class

I and II rules as they are visually easy to identify. The remaining rules tend to have more ambiguity. We envision trying to place ambiguous ECAs on a *spectrum* with one end being clear Class III rules while the other end having clear Class IV rules. These two extremes are satisfied with these well-established rules:

Rule 30 It is used for generating pseudo-random numbers in Mathematica [112, 113].

Rule 30 is an iconic example of a chaotic ECA in Class III.

Rule 110 It has been proven to be capable of universal computation as information can be encoded into ‘gliders’ [114]. Rule 110 is an iconic example of a complex ECA in Class IV.

There exists ambiguous rules whose illustrative examples include:

Rule 54 This rule exhibits a glider system where gliders are observed to interact.

Rule 54’s glider system has been studied in detail [115–117] but its capacity for universal computation is unknown.

Rule 18 Initially assigned by Wolfram to Class III, similar to Rule 30, though subsequent studies indicate anomalies within the ECA called kinks (see Fig. 7.2) that can propagate and annihilate other kinks [118, 119]. This indicates some level of structure.

Rule 18, featured in Wolfram’s Class III, contains a phenomenon that has been referred to as ‘kinks’ [118]. Kinks are identified with the presence of two adjacent black cells in the ECA state, and have been shown capable of undergoing random walks [119]. Upon meeting, kinks annihilate. A filter can be applied to the dynamics of Rule 18 by replacing all cells with white cells with the exception of adjacent pairs of black cells. The movement and interaction of kinks is reminiscent to that of a glider systems in Rules 54 and 110.

Figure 7.2: Rule 18 with and without a filter. The filter indicates the location of the kinks.

Since computational mechanics has been used to quantify structure that a process contains (see interpretations of statistical complexity in Chapter 3.1), the theme of this Chapter uses the quantum statistical memory to quantify structure in ECA. Therefore, we ask the question:

Can the quantum statistical memory offer a new method to identify and classify complex ECA?

We consolidate our findings on understanding the behaviour of various ambiguous ECA rules through the lens of quantum computational mechanics, specifically using the Quantum Inference Protocol.

To do so, we first show that the quantum statistical memory $\tilde{C}_q$ is able to identify well-established rules in ECA. Further investigation of various rules indicate that certain seemingly-chaotic ECA rules are able to generate structure, hence we call them ambiguous rules. Our method of applying quantum computational mechanics allows a refinement of Class III and Class IV rules based on how their relative complexity evolves.

To our knowledge, our proposed classification scheme is the first of its kind to offer a hybrid mix of a continuous (fine graining) analysis of chaotic and complex rules together with a discrete classification of trivial, periodic, pseudo-random rules.

7.3 A stochastic perspective

The tools of computational mechanics, established in Chapter 3.1, were used to classify ECA, allowing ambiguous ECA to be placed on a spectrum with one extreme being fully random and the other extreme being fully complex. To do so, we have to describe ECA in the language of stochastic processes.

An ECA is initialised with random initial conditions. This corresponds to the ECA at $t = 0$ being described by a spatially-stationary stochastic process $\vec{X}^{(0)}$ that is completely random with $\tilde{C}_q^{(t)} = 0$. Allowing the ECA to act on the initial condition can be seen as transforming the initial condition into a new stochastic process, which we denote with $\vec{X}^{(t)}$. The update rules are translationally invariant, implying that the resulting $\vec{X}^{(t)}$ are also spatially-stationary.

We can therefore compute a well-defined quantum statistical memory of $\vec{X}$. Observing how $C_q^{(t)}$ evolves with t allows us to quantify the amount of structure created or destroyed by the ECA's evolution. To this end, we propose the following complexity criterion.

7.4 Complexity criterion

A complexity criterion

An ECA is complex if the quantum statistical memory, $C_q^{(t)}$, grows with t without bound.

The basis of the criterion hinges on the fact that complex dynamics should be capable of generating new structure. Building up complex correlations between cells arbitrarily far apart as t grows, requiring us to track ever more information. This implies that simple ECA are termed as simple as they generate little structure and

thus the exact quantum statistical memory $C_q^{(t)}$ plateaus. We can therefore make the following statements regarding simple ECA (Class I, II, and III):

- (1) $\lim_{t \rightarrow \infty} C_q^{(t)} = 0$ for all Class I ECA.
- (2) $C_q^{(t)}$ must be asymptotically bounded for all Class II ECA. There exists a sufficiently large time T and some constant K such that $C_q^{(t)} \leq K$ for all $t > T$.
- (3) $C_q^{(t)} \approx 0$ for ECA suitable for use as a near-perfect pseudo-random number generators.

Statement (1) follows as all cells in the ECA asymptote to homogeneous states after sufficiently large t . The resultant stochastic process has been transduced by from a random process to fully ordered process by Class I ECA rules. Since a fully ordered process only has a single causal state, $C_q^{(t)} = 0$.

Statement (2) revolves around the fact that Class II ECAs asymptote to a periodic configuration with periodicity τ (modulo spatial translations as spatial translations do not alter C_q .) At the asymptotic periodic state, $C_q^{(t)}$ oscillates with periodicity τ and is upper bounded by some maximal constant K .

Lastly, on Statement (3), a perfect random number generator should generate no correlations between each outcome and should result in $C_q^{(t)} = 0$. Rule 30 is used as a pseudo random number generator in Mathematica [92, 112] and hence should result in no correlations between time steps in the ECA. The continuity of C_q (see Chapter 4.4) ensures that $C_q^{(t)} \approx 0$.

7.5 The truncated methodology

The ambiguous ECAs that are of interest are those that have a potential computational irreducibility; there is no clear-cut method of determining the long term behaviour

Figure 7.3: Our workflow to obtain the quantum statistical memory $\tilde{C}_q^{(t)}$ of the ECA at each time step. For completeness, we use the classical sub-tree algorithm to infer $\tilde{C}_\mu^{(t)}$ of the ECA at each time step.

of $C_q^{(t)}$ just like for Class I and II ECAs. As such, numerical methods are employed to better understand the long term behaviour of the ECA. The brief methodology for carrying out numerical simulations is illustrated in Fig. 7.3 while the extended methodology can be found in Appendix A.1.

In Appendix A.1, we dive into a short discussion about the difference between ECAs that have periodic boundary conditions and no periodic boundary conditions (also known as open boundaries). Since the numerical results show little difference, we use ECAs with no periodic boundary conditions throughout this Chapter and leave the discussion on boundary conditions to the Appendix.

Each ECA is generated with $W = 64,000$ cells such that the state of the ECA at any time t can be described by a finite data string $x_{0:W}^{(t)}$. We then make use of the Quantum Inference Protocol to estimate the quantum statistical complexity $\tilde{C}_q^{(t)}$ at every time step. The maximum time step we consider is $t_{\max} = 10^3$ to capture most of the ECA's dynamics before they reach a pseudo-steady state. For completeness, the classical statistical complexity $C_\mu^{(t)}$ is computed for all 88 independent rules as well, and its algorithm is based on the sub-tree algorithm found in Chapter 3.1.1. The results can be found in Fig. 7.5 and 7.6.

7.6 Numerical Results

Our results are summarised with a figure containing ambiguous ECAs that reside in Wolfram's Class III and IV. By plotting $\tilde{C}_q^{(t)}$ as a function of time, we observe that the extremal cases behave as expected:

Rule 30 It is observed that $\tilde{C}_q^{(t)} \approx 0$, implying that there is no structure generated from the (random) initial conditions. This highlights the consistence with its role as a pseudo-random number generator.

Rule 110 At the other end of the spectrum, Rule 110 portrays the fastest growth in $\tilde{C}_q^{(t)}$. Because Rule 110 is proven to be capable of universal computation, it is implied that it can propagate correlations over arbitrarily large distances.

With both extremal cases having their numerics computed and verified against our complexity criterion, we can now turn our attention to the ambiguous rules that reside between the extremes. We list them in order of increasing $\tilde{C}_q^{(t)}$ rates.

Rule 22 We see that $\tilde{C}_q^{(t)}$ increases but stagnates in growth after a short while. This hints at Rule 22 being a simple ECA according to the complexity criterion; Rule 22's $\tilde{C}_q^{(t)}$ does not grow without bound, and so it is a simple ECA. Prior studies have indicated that Rule 22 behaves as a mix between random and periodic process, generating a finite amount of local structure that does not propagate with time. As such, it is very unlikely to be Turing complete [120].

Rule 18 Despite being an ECA that is classified as a chaotic rule, Rule 18 exhibits a slow growth in $\tilde{C}_q^{(t)}$. The growth in $\tilde{C}_q^{(t)}$ suggests the presence of some complex dynamics. It was established that Rule 18 has kinks [118, 119] that are able to interact and annihilate, supporting the observation of some complex dynamics. Rule 18 and Rule 146 have very similar growth rates.

Figure 7.4: A spectrum of ECA complexity with Rule 30 (Class III, simple) and Rule 110 (Class IV, complex) at the extremes. On each plot, the lines represent the average $\tilde{C}_q^{(t)}$ from 5 random initial conditions. Error bars are present but too small to be visible at scale shown. Rules 22, 18, and 122 are Wolfram Class III while Rule 54 is Class IV – our spectrum is consistent with this division, but shows further nuance. Plots for all rules can be found in the Supplementary Material.

Rule 122 The growth in $\tilde{C}_q^{(t)}$ is similar to that of Rule 18 but much faster. Rule 122 and Rule 126 have very similar growth rates.

Rule 54 This is the only ECA that was classified by Wolfram to be in Class IV. Indeed, studies have documented a glider system with gliders that are able to interact [115–117] but is noted to be less complex than Rule 110 [121]. At the time of writing, Rule 54's capability to perform universal computation remains to be proven.

Apart from Rule 22 which has a stagnating $\tilde{C}_q^{(t)}$ and is regarded as a simple rule, the remaining ambiguous rules listed above do feature some degree of underlying complex dynamics. As such, having a spectrum and ranking their relative growth rates in $\tilde{C}_q^{(t)}$ appears to offer a suitable method of understanding their underlying dynamics. This provides a greater refinement to the discrete classifications that Wolfram proposed.

For completeness we plot all 88 independent ECAs with open boundary conditions coloured according to Wolfram's classification and demonstrate how Wolfram's Class

III and IV rules can be plotted on a spectrum just like in Fig. 7.4. Here, we include $\tilde{C}_\mu^{(t)}$ (red plots) and $\tilde{C}_q^{(t)}$ (blue plots) results.

Figure 7.5: Class I and II.

Figure 7.6: Class III and IV plotted on a continuous spectrum.

7.7 Discussion

In this Chapter, we viewed one-dimensional cellular automata as a transducer from one stochastic process to another. The ECA were initialised with random initial conditions with no structure and allowed to evolve under various ECA rules. We postulated that various ECA rules should be able to generate varying degrees of structure which inspired the complexity criterion. It states that complex rules should be able to generate increasing amounts of structure while simple rules do not generate structure – or have stagnating structure, if any at all.

Numerical methods employed in this Chapter use the Quantum Inference Protocol to obtain the quantum statistical memory of the ECA at each time step. This has allowed us to quantitatively ascertain two extremal cases of ECA which we expect: (i) ECA (Rule 30) that is used for pseudo-random number generation is indeed true; no structure is generated at all; and (ii) ECA (Rule 110) that is capable of universal computation should be able to generate increasing complex behaviour due to their ability to propagate correlations over large distances. Our numerical observations also point to certain ambiguous Class III ECA having the capability to generate some degree of complex behaviour. These ambiguous ECA that can be placed onto a spectrum between the two above-mentioned extremes.

Our methodology has allowed us to classify ECA according to a hybrid of discrete classes while offering a further refinement to Class III and IV ECA into a continuous spectrum. Our results agree with prior literature when there is a consensus on the perceived complexity of ECA, wherein ‘complexity’ in this scenario refers to something that is generally hard to describe.

There is interest in making modifications to the basic ECA in hopes of allowing modified rules to generate structure, such as those with more cell states (black, white, grey, etc), longer-range update rules, asynchronous tuning [122], and memory [123].

Our framework provides a stepping stone for new insight into what is complex in these less-extensively studied situations.

CHAPTER 8

Conclusion and future works

*There is no real ending.
It's just the place where you stop the story.*

- Frank Herbert

This thesis is a testament of how fortunate we are to be at the intersection between two scientific fronts — complexity science, and quantum computation. It is a culmination of work starting from the Quantum Inference Protocol, branching out into various directions from unitary construction, gate decomposition, error mitigation, to quantifying structure induced by elementary cellular automata rules on stochastic processes. The conclusion of this thesis marks the beginning of advancing the known boundaries of science.

Different areas of research are similar to rivers. A new path for water to flow is carved out when two streams meet but may split into various branches downstream.

Figure 8.1: Two streams of water meet and form a new stream. The new stream may also branch out into different streams. Replace the word 'stream' with 'scientific topic'. Illustration drawn by the author of this thesis.

The study of quantum causal models prior to the Quantum Inference Protocol is like a stream after two rivers join together, with the two rivers being computational mechanics and quantum mechanics. The Quantum Inference Protocol is just one of the streams that have been split from the main study of quantum causal models, but is also able to branch out into different streams itself. New research ideas are formed from past work, thereby pushing the frontier of science.

Figure 8.2: The thesis can be summarised with this workflow that traces the origins of the Quantum Inference Protocol as well as how it branches out into various fields of research.

Before more illustrations turns into a bedtime story, let us retrace our steps that led us to the Quantum Inference Protocol by looking at historical research laid out in Chapter 3. Causal models based on computational mechanics were introduced in Chapter 3 to analyse complex stochastic processes. Instead of remembering all possible pasts, computational mechanics groups pasts that have statistically identical futures together in an equivalence class. Each past leads to some future, implying that each equivalence class leads to another equivalence class with some transition output. After establishing the equivalence classes and the transitions, one would only need to track which equivalence classes the process is in to be able to chart out a valid statistical future. The model which computational mechanics produces is called the ε -machine. Running the ε -machine step by step produces a stochastic process that is statistically distributed according to the input stochastic process. The statistical complexity, which indicates the memory requirements of ε -machines, solely

depend on the stationary probability distribution of equivalence classes.

Quantising each classical equivalence class into quantum states gives the set of quantum states. These quantum states have the potential to exist non-orthogonally with each other, allowing a reduction in the memory requirement but retaining predictive capabilities. We refer to the memory requirement as the quantum statistical memory. There are three main versions in total. The first version, despite seeming like the prototype in this thesis, displayed a clear edge over classical models; its quantum statistical memory is smaller than the statistical complexity. The second iteration showed that increasing the futures encoded in the quantum states allowed the non-orthogonality to increase up to the cryptic order, further minimising the quantum statistical memory. The state-of-the-art quantum causal models are unitary quantum causal models. Unitary quantum models were reframed in quantum computation framework in preparation for quantum simulation. Its successor features complex phases and is currently the best quantum causal model which minimises C_q the most. However, the phase-dependent unitary quantum models lack a systematic method of obtaining the optimal complex phases that are specific to each set of quantum memory states. One potential area of development is to engage machine learning techniques to extract these optimal complex phase that leads to the lowest possible quantum statistical memory.

Unfortunately all aforementioned quantum causal models are reliant on the ε -machine's construction. The set of equivalence classes is needed for the one-to-one mapping to quantum states. Problems arise when dealing with real world finite stochastic processes. The construction of the equivalence classes in the ε -machine requires a δ -tolerance parameter to account for statistical fluctuations arising from finite processes. A wrong choice of δ -parameter produces inaccurate ε -machines.

The implication is that the accuracy of the aforementioned quantum causal models are also dependent on the δ -tolerance parameter. As such, the Quantum Inference Protocol (Chapter 4) was developed to resolve this issue.

The Quantum Inference Protocol produces quantum memory states through the direct inference of the stochastic processes without a need for the δ -tolerance. The Quantum Inference Protocol's quantum statistical memory was shown to be similar to the exact quantum statistical memory as long as sufficient data is provided and the length of past to observe is at least the Markov order of the process (i.e. optimal history length). The Quantum Inference Protocol was theoretically validated by showing that self-merging of quantum memory states occurs at and above the Markov order as well as the continuity of the von Neumann entropy. On the practical front, we introduced the concept of effective Markov order which one could use in place of the optimal history length should the Markov order be unknown or infinite. In fact, the Quantum Inference Protocol also allows one to quickly estimate the amount of memory required for phaseless unitary quantum models to model a stochastic process.

With NISQ platforms becoming increasingly popular to conduct one's own experiments, especially since these devices are reachable via the cloud ('cloud' referring to cloud computing), the avenue of applying the Quantum Inference Protocol in real quantum devices opened up. We sought to use the Quantum Inference Protocol to predict the future of stochastic processes. The methodology for constructing the unitary operator was codified and a separate algorithm was written to decompose the unitary operator to elementary quantum gates that are implementable on quantum devices. To ensure that the unitary operator (and hence the number of elementary quantum gates) is within implementable means, the inferred quantum memory states from the Quantum Inference Protocol were merged if the fidelity between two states

was at least within δ -tolerance. The recommended δ -tolerance depends on N , the length of the stochastic process, and is analytically found to be roughly $\delta = \frac{1}{2\sqrt{N}}$.

The quantum devices of this era are inherently noisy, hence the name “Noisy Intermediate-Scale Quantum” (NISQ) devices [26]. Noise affects measurement outcomes by skewing the expected values away from the exact values. Simulating the Quantum Inference Protocol on real quantum devices means that our results inherit the effects of noise. In Chapter 6, we use error mitigation methods to circumvent the effects of noise. More specifically, we use the probabilistic cancellation method. We showed that the cumulative sampling cost for error mitigation is related to how noisy the device is, which generally puts our quantum models of stochastic processes at a disadvantage over classical models. However, there is a narrow region where quantum memory states have sufficient overlaps to allow error mitigated quantum models to have an advantage over classical models. We expect this region to broaden as quantum technology improves, in turn driving the cumulative sampling cost down.

Potential areas of exploration for error mitigation revolve around reducing the C -coefficient (which indirectly represents the sampling cost) is to use a decomposition method with smaller circuit depth; a smaller circuit depth indicates less quantum gates which implies that noise has a smaller effect on. In addition, different basis operations can be used for quantum error mitigation. Potential future work can be focused on exploring the use of a continuous set of noisy implementable operations [81–85] which may reduce the C -coefficient. Another potential outlook is to explore if error mitigation methods may be applied to overcome noise due to crosstalk [124].

We mentioned the various interpretations of the statistical complexity in Chapter 3.1. Apart from operational meaning, both the classical statistical complexity and

quantum statistical memory also indicate the amount of structure a stochastic process contains. This interpretation of the quantum statistical memory was used to identify structure within elementary cellular automata (ECA). Generally, cellular automata is basically a collection of cells that evolve based on specific permutations of neighbouring cells and its past cell that are called rules. Elementary cellular automata is the simplest form of cellular automata with a one-dimensional chain of cells. By applying rules to a chain of ECA cells, these rules will evolve the ECA to show various types of behaviour. For a chain of cells that are initially randomly configured, various classification schemes emerged, with the most notable one being the Wolfram classification scheme that featured four classes—simple, periodic, chaotic, complex.

By using the Quantum Inference Protocol to find the quantum statistical memory of the ECA at each time step after the rule is applied, we are able to observe if any structure evolves out of randomness for each ECA rule. Our results show that the ECA rules can be classified according to how much structure each rule contains. Our classification scheme aligns mostly with Wolfram’s four classes but offers a refinement to the chaotic and complex classes; there exists a continuous spectrum between chaotic and complex classes where ECA rules fall on. Therefore, our work sheds light on how ECA rules can be classified in a hybrid manner between discrete and continuous classes.

As with any other piece of research work that hinges on past research ideas, this thesis demonstrates that we are only scratching the tip of the iceberg. Indeed, the future research trajectories of this now-historical thesis are endless, however small or large its impact. The quote at the beginning of this Chapter, though cliché and overused in many conclusions in theses, books, and writings, embodies the epitome of scientific research. Any piece of research work, just like any of the chapters in this

thesis, will always push the scientific frontier forward into the unknown.

Appendix A

ECA Methodology

A.1 Methodology for ECA

In this work, we study finite-width analogues of ECA. To avoid boundary effects from the edges of the ECA of width W for up to $t_{\max}$ timesteps, we generate an extended ECA of width $W' = W + 2t_{\max}$ that has periodic boundary conditions and keep only the centremost W cells. The resulting ECA will have width W that has open boundaries (see Fig. A.1). Note however that the choice of boundary condition showed little quantitative effect upon our results.

Figure A.1: Generation of finite-width ECA evolution with open boundaries via extended ECA with periodic boundaries.

The ECA at each time step can be interpreted as a stochastic process. That is, given an ECA at time t in state $x_{0:W}^{(t)}$, we can interpret this as a finite string

of outcomes from a stochastic process $y_{0:W}^{(t)}$ with the same alphabet. The tools of computational mechanics can then be applied to this finite string, inferring the classical statistical complexity C_μ through the sub-tree reconstruction algorithm, and the quantum statistical memory $\tilde{C}_q$ from the quantum inference protocol. For both inference methods we use $L = 6$, as little qualitative difference was found using larger L (see Fig. A.3). For the sub-tree reconstruction algorithm, tolerance of the χ^2 test was set to 0.05.

We apply the inference algorithms to ECA states of width $W = 64,000$. For each ECA rule, we generate an initial state for $t = 1$ where each cell is randomly assigned 0 or 1 with equal probability, and then evolve the ECA for $t_{\max} = 10^3$ steps (in Fig. A.3 under the ECA rules. We stop at $t_{\max} = 10^3$ as selected rules simulated for up to $t_{\max} = 10^5$ show that the qualitative features of interest are already captured at the shorter number of timesteps). Note that with $W = 64,000$, $t_{\max} = 10^3$ gives many orders of magnitude smaller than the time for which a typical finite-width ECA is guaranteed to cycle through already-visited states ($\mathcal{O}(2^W)$) [120]. We then apply the inference methods to the states at these timesteps $t = 1, 2, 3, \dots, 9, 10, 20, \dots, 90, 100, 200, \dots, t_{\max}$. Evaluating $C_q^{(t)}$ and $C_\mu^{(t)}$ at every timestep shows little qualitative difference beyond highlighting the short periodicity of some Class II rules. We repeat five times for each rule, essentially simulating each ECA with five random initial conditions, and plot the mean and standard deviation of $C_q^{(t)}$ and $C_\mu^{(t)}$. In our plots, the standard deviation of $C_q^{(t)}$ is small compared to the vertical axis.

A.2 Boundary conditions

Figure A.2: Four rules (each from Wolfram's classes) are used to illustrate that there is little quantitative difference between ECAs with periodic boundary conditions and ECAs with open boundary conditions.

A.3 Longer L , larger $t_{\max}$

In Fig. [A.3](#) we present plots supporting that our choice of $L = 6$, $t_{\max} = 10^3$ appears to be sufficiently large enough to capture the qualitative features of ECA evolution, showing little difference when both L and $t_{\max}$ are extended.

Figure A.3: $C_q^{(t)}$ plots for a selection of rules with longer L and larger $t_{\max}$. Plots shown for $\{W = 64,000, L = 6\}$, $\{W = 128,000, L = 7\}$, and $\{W = 256,000, L = 8\}$.

Rule 110 appears to be the only exception as it plateaus at longer times. We believe this is attributable to the finite width of ECA. A finite width of ECA begets finite number of gliders generated by the initial configuration. Over time, the number of gliders annihilating outweighs the number of gliders that do not annihilate or reproduce, thus there will be fewer gliders left to interact and propagate further correlations. This can be seen in Fig [A.4](#)

Figure A.4: Over time, the finite number of gliders present in the initial configuration of a finite-width Rule 110 ECA tends to disappear due to annihilation with other gliders. At longer times, there are then fewer gliders to propagate longer-range correlations across the ECA.

Bibliography

- [1] *What is QLED and Neo QLED? Samsung's TV tech explained.* url: <https://www.pocket-lint.com/tv/news/samsung/139867-what-is-qled-tv-samsung-television-technology-explained> (visited on 03/15/2021).
- [2] *Samsung and SK Telecom reveal world's first smartphone with quantum security tech.* url: <https://www.engadget.com/samsung-and-sk-telecom-reveal-worlds-first-smartphone-with-quantum-security-tech-143049380.html> (visited on 03/15/2021).
- [3] *Samsung's Galaxy Quantum 2 has quantum cryptography built in.* url: <https://www.theverge.com/2021/4/13/22381321/samsung-galaxy-quantum-2-announced-qrng-cryptography-chip> (visited on 04/16/2021).
- [4] *Quantum Of Solace.* url: <https://www.mgm.com/movies/quantum-of-solace> (visited on 03/15/2021).
- [5] *Just What IS a "Quantum"?* url: <http://physicsbuzz.physicscentral.com/2018/02/just-what-is-quantum.html> (visited on 03/15/2021).
- [6] P W Shor. "Algorithms for quantum computation: discrete logarithms and factoring". *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134.
- [7] Lov K. Grover. "Quantum Mechanics Helps in Searching for a Needle in a Haystack". *Physical Review Letters* 79, 325–328 (1997).
- [8] Charles H. Bennett and Gilles Brassard. "Quantum cryptography: Public key distribution and coin tossing.pdf". *Proceedings of the International Conference on Computers, Systems and Signal Processing, Bangalore, India, 1984*. 1984, pp. 175–179.
- [9] Artur K Ekert. "Quantum cryptography based on Bell's theorem". *Physical Review Letters* 67, 661–663 (1991).
- [10] Nicolas Gisin et al. "Quantum cryptography". *Reviews of Modern Physics* 74, 145–195 (2002).
- [11] James P. Crutchfield and Karl Young. "Inferring statistical complexity". *Physical Review Letters* 63, 105–108 (1989).
- [12] James P Crutchfield. "Between order and chaos". *Nature Physics* 8, 17–24 (2011).

- [13] C R Shalizi. "Causal Architecture, Complexity and Self-Organization in Time Series and Cellular Automata". *PhD Thesis* (2001). url: <http://bactra.org/thesis/>.
- [14] Cosma Rohilla Shalizi and Kristina Lisa Klinkner. "Blind Construction of Optimal Nonlinear Recursive Predictors for Discrete Sequences". *Uncertainty in Artificial Intelligence: Proceedings of the Twentieth Conference (UAI 2004)*. Ed. by Max Chickering and Joseph Y. Halpern. Arlington, Virginia: AUAI Press, 2004, pp. 504–511.
- [15] W. M. Gonçalves et al. "Inferring statistical complexity in the dripping faucet experiment". *Physica A* 257, 385–389 (1998).
- [16] Joongwoo Brian Park et al. "Complexity analysis of the stock market". *Physica A* 379, 179–187 (2007).
- [17] Robert Haslinger, Kristina Lisa Klinkner, and Cosma Rohilla Shalizi. "The computational structure of spike trains". *Neural Computation* 22, 121–157 (2010).
- [18] Roberto N Muñoz et al. "General anesthesia reduces complexity and temporal asymmetry of the informational structures derived from neural recordings in *Drosophila*". *Phys. Rev. Research* 2, 23219 (2020).
- [19] Mile Gu et al. "Quantum mechanics can reduce the complexity of classical models". *Nature Communications* 3, 762 (2012).
- [20] Ryan Tan et al. "Towards quantifying complexity with quantum mechanics". *European Physical Journal Plus* 129:191 (2014).
- [21] John R Mahoney, Cina Aghamohammadi, and James P Crutchfield. "Occam's Quantum Strop: Synchronizing and Compressing Classical Cryptic Processes via a Quantum Channel". *Scientific Reports* 6, 20495 (2016).
- [22] Paul M. Riechers et al. "Minimized state complexity of quantum-encoded cryptic processes". *Physical Review A* 93, 052317 (2016).
- [23] Felix C. Binder, Jayne Thompson, and Mile Gu. "A Practical Unitary Simulator for Non-Markovian Complex Processes". *Physical Review Letters* 120, 240502 (2018).
- [24] Qing Liu et al. "Optimal stochastic modeling with unitary quantum dynamics". *Physical Review A* 99, 062110 (2019).
- [25] Matthew Ho, Mile Gu, and Thomas J Elliott. "Robust inference of memory structure for efficient quantum modeling of stochastic processes". *Physical Review A* 101, 032327 (2020).
- [26] John Preskill. "Quantum Computing in the NISQ era and beyond". *Quantum* 2, 79 (2018).
- [27] John Preskill. "Quantum computing and the entanglement frontier". *arXiv*, 1–18 (2012). arXiv: [1203.5813](https://arxiv.org/abs/1203.5813).

- [28] Frank Arute et al. “Quantum supremacy using a programmable superconducting processor”. *Nature* 574, 505–510 (2019).
- [29] A. M. Steane. “Active stabilization, quantum computation, and quantum state synthesis”. *Physical Review Letters* 78, 2252–2255 (1997).
- [30] Daniel Gottesman. “Stabilizer Codes and Quantum Error Correction”. *PhD Thesis* 2008 (1997). arXiv: [9705052 \[quant-ph\]](#).
- [31] Andrew M. Steane. “Fast fault-tolerant filtering of quantum codewords” (2002). arXiv: [0202036 \[quant-ph\]](#).
- [32] Daniel Gottesman. “An introduction to quantum error correction”. *arXiv*, 221–235 (2002). arXiv: [0004072 \[quant-ph\]](#).
- [33] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. “Error Mitigation for Short-Depth Quantum Circuits”. *Physical Review Letters* 119, 180509 (2017).
- [34] Ying Li and Simon C Benjamin. “Efficient Variational Quantum Simulator Incorporating Active Error Minimization”. *Physical Review X* 7, 021050 (2017).
- [35] Jarrod R McClean et al. “Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states”. *Physical Review A* 95, 042308 (2017).
- [36] X. Bonet-Monroig et al. “Low-cost error mitigation by symmetry verification”. *Physical Review A* 98, 062339 (6 2018).
- [37] Suguru Endo, Simon C. Benjamin, and Ying Li. “Practical Quantum Error Mitigation for Near-Future Applications”. *Physical Review X* 8, 031027 (2018).
- [38] Sam McArdle, Xiao Yuan, and Simon Benjamin. “Error-Mitigated Digital Quantum Simulation”. *Physical Review Letters* 122, 180501 (2019).
- [39] Bálint Koczor. “Exponential Error Suppression for Near-Term Quantum Devices”. *arXiv* (2021). arXiv: [2011.05942 \[quant-ph\]](#).
- [40] William J Huggins et al. “Virtual Distillation for Quantum Error Mitigation”. *arXiv* (2021). arXiv: [2011.07064 \[quant-ph\]](#).
- [41] Matthew Ho, Ryuji Takagi, and Mile Gu. “Enhancing quantum models of stochastic processes with error mitigation”. *arXiv* (2021). arXiv: [2105.06448](#).
- [42] Matthew Ho et al. “Quantum-inspired identification of complex cellular automata”. *arXiv* (2021). arXiv: [2103.14053](#).
- [43] Shu Hotta. “Hermitian Operators and Unitary Operators”. *Mathematical Physical Chemistry: Practical and Intuitive Methodology*. Singapore: Springer Singapore, 2018, pp. 409–441.
- [44] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. 2010.
- [45] Daniel Greenbaum. *Introduction to Quantum Gate Set Tomography*. 2015. arXiv: [1509.02921 \[quant-ph\]](#).

- [46] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. 1991.
- [47] Peter Lynch. “The origins of computer weather prediction and climate modeling”. *Journal of Computational Physics* 227, 3431–3444 (2008).
- [48] Peter Bauer, Alan Thorpe, and Gilbert Brunet. “The quiet revolution of numerical weather prediction”. *Nature* 525, 47 (2015).
- [49] “Measuring complexity, nonextensivity and chaos in the DNA sequence of the Major Histocompatibility Complex”. *Physica A* 438, 188–209 (2015).
- [50] Tobias Preis et al. “Quantifying the behavior of stock correlations under market stress”. *Scientific Reports* 2, 1–5 (2012).
- [51] Andreas Schadschneider, Debashish Chowdhury, and Katsuhiro Nishinari. *Stochastic Transport in Complex Systems: From Molecules to Vehicles*. Elsevier, 2010.
- [52] William, Philotheus. Boehner, and Stephen F Brown. *Philosophical writings: a selection / William of Ockham; translated, with introduction and notes, by Philotheus Boehner; Latin texts and English translation revised by Stephen F. Brown, 1989; new foreword and bibliography by Stephen F. Brown*. Indianapolis: Hackett Pub. Co., 1990.
- [53] Erich Becher. “The Philosophical Views of Ernst Mach”. *The Philosophical Review* 14, 535–562 (1905).
- [54] Paul M. Riechers and James P. Crutchfield. “Spectral simplicity of apparent complexity. I. the nondiagonalizable metadynamics of prediction”. *Chaos* 28 (2018).
- [55] Paul M. Riechers and James P. Crutchfield. “Spectral simplicity of apparent complexity. II. Exact complexities and complexity spectra”. *Chaos* 28 (2018).
- [56] Whei Yeap Suen et al. “The classical-quantum divergence of complexity in modelling spin chains”. *Quantum* 1, 25 (2017).
- [57] Christopher C. Strelhoff and James P. Crutchfield. “Bayesian structural inference for hidden processes”. *Physical Review E* 89, 042119 (2014).
- [58] John R. Mahoney et al. “How hidden are hidden processes? A primer on crypticity and entropy convergence”. *Chaos* 21, 1–18 (2011).
- [59] E. Racca et al. “Test to determine the Markov order of a time series”. *Physical Review E* 75, 011126 (2007).
- [60] Richard Jozsa. “Distinguishability of states and von Neumann entropy”. *Physical Review A* 62, 012301 (2000).
- [61] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. 2nd ed. Cambridge University Press, 2012.
- [62] Charles Van Loan. “Computing the CS and the generalized singular value decompositions”. *Numerische Mathematik* 46, 479–491 (1985).

- [63] Zhaojun Bai and James W Demmel. “Computing the Generalized Singular Value Decomposition”. *SIAM Journal on Scientific Computing* 14, 1464–1486 (1993).
- [64] C. C. Paige and M. Wei. “History and generality of the CS decomposition”. *Linear Algebra and Its Applications* 208–209, 303–326 (1994).
- [65] Mikko Möttönen et al. “Quantum Circuits for General Multiqubit Gates”. *Physical Review Letters* 93, 130502 (2004).
- [66] Mikko Möttönen and Juha Vartiainen. “Decompositions of general quantum gates”. *Frontiers in Artificial Intelligence and Applications* (2005).
- [67] Y. G. Chen and J. B. Wang. “Qcompiler: Quantum compilation with the CSD method”. *Computer Physics Communications* 184, 853–865 (2013).
- [68] Yumi Nakajima, Yasuhito Kawano, and Hiroshi Sekigawa. “A new algorithm for producing quantum circuits using KAK decompositions”. *Quantum Information and Computation* 6, 067–080 (2006).
- [69] Hartmut Führ and Ziemowit Rzeszutnik. “A note on factoring unitary matrices”. *Linear Algebra and its Applications* 547, 32–44 (2018).
- [70] *Two Qubit Basis Decomposer*. url: https://qiskit.org/documentation/stubs/qiskit.quantum_info.TwoQubitBasisDecomposer.html (visited on 03/30/2021).
- [71] A. De Vos and S. De Baerdemacker. “Block-ZXZ synthesis of an arbitrary quantum circuit”. *Physical Review A* 94, 052317 (2016).
- [72] Matthew S Palsson et al. “Experimentally modeling stochastic processes with less memory by the use of a quantum processor”. *Science Advances* 3 (2017).
- [73] Farzad Ghafari Jouneghani et al. “Observing the ambiguity of simplicity via quantum simulations of an Ising spin chain”. *arXiv* (2017). arXiv: [1711.03661](https://arxiv.org/abs/1711.03661).
- [74] Farzad Ghafari et al. “Single-shot quantum memory advantage in the simulation of stochastic processes”. *arXiv* (2018). arXiv: [1812.04251](https://arxiv.org/abs/1812.04251).
- [75] Andrew W. Cross et al. “Open quantum assembly language”. *arXiv* (2017). arXiv: [1707.03429](https://arxiv.org/abs/1707.03429).
- [76] Cirq Developers. *Cirq*. 2021. url: <https://doi.org/10.5281/zenodo.4586899>.
- [77] J R Johansson, P D Nation, and Franco Nori. “QuTiP: An open-source Python framework for the dynamics of open quantum systems”. *Computer Physics Communications* 183, 1760–1772 (2012).
- [78] J R Johansson, P D Nation, and Franco Nori. “QuTiP 2: A Python framework for the dynamics of open quantum systems”. *Computer Physics Communications* 184, 1234–1240 (2013).
- [79] Chao Song et al. “Quantum computation with universal error mitigation on a superconducting quantum processor”. *Science Advances* 5 (2019).

- [80] Seth T Merkel et al. "Self-consistent quantum process tomography". *Physical Review A* 87, 062119 (2013).
- [81] Ryuji Takagi. "Optimal resource cost for error mitigation". *Phys. Rev. Research* 3, 033178 (2021).
- [82] Jiaqing Jiang, Kun Wang, and Xin Wang. "Physical Implementability of Quantum Maps and Its Application in Error Mitigation". *arXiv* (2020). arXiv: [2012.10959 \[quant-ph\]](#).
- [83] Bartosz Regula, Ryuji Takagi, and Mile Gu. "Operational applications of the diamond norm and related measures in quantifying the non-physicality of quantum maps". *arXiv* (2021). arXiv: [2102.07773 \[quant-ph\]](#).
- [84] Yifeng Xiong et al. "Sampling overhead analysis of quantum error mitigation: Uncoded vs. coded systems". *arXiv* (2020). arXiv: [2012.08378](#).
- [85] Christophe Piveteau, David Sutter, and Stefan Woerner. "Quasiprobability decompositions with reduced sampling overhead". *arXiv* (2021). arXiv: [2101.09290](#).
- [86] Richard Jozsa. "Fidelity for Mixed Quantum States". *Journal of Modern Optics* 41, 2315–2323 (1994).
- [87] Elwyn R Berlekamp. *Winning ways for your mathematical plays. Volume 2*. Ed. by John H (John Horton) Conway and Richard K Guy. Boca Raton, FL, 2018.
- [88] Paul Rendell. "Turing machine in Conway game of life". *Designing Beauty: The Art of Cellular Automata*. Springer, 2016, pp. 149–154.
- [89] Stephen Wolfram. "Statistical mechanics of cellular automata". *Reviews of Modern Physics* 55, 601–642 (1983).
- [90] Stephen Wolfram. "Universality and complexity in cellular automata". *Physica 10D*, 1–35 (1984).
- [91] Stephen Wolfram. "Universality and Complexity in Cellular Automata, Cellular Automata and Complexity: Collected Papers". *Physica D* 10, 140–157 (1995).
- [92] Stephen Wolfram. *A New Kind of Science*. Vol. 56. Applied Mechanics Reviews, 2003, p. 1197.
- [93] Wentian Li and N. H. Packard. "The Structure of the Elementary Cellular Automata Rule Space". *Complex Systems* 4, 281–297 (1990).
- [94] Stephen Wolfram. "Cellular automata as models of complexity". *Nature* 311, 419–424 (1984).
- [95] Shigeru Ninagawa. "Power Spectral Analysis of Elementary Cellular Automata". *Complex Systems* 17, 399–411 (2008).

- [96] Eurico L P Ruivo and Pedro P B de Oliveira. "A Spectral Portrait of the Elementary Cellular Automata Rule Space". *Irreducibility and Computational Equivalence: 10 Years After Wolfram's A New Kind of Science*. Ed. by Hector Zenil. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 211–235.
- [97] Christopher G. Langton. "Studying artificial life with cellular automata". *Physica D: Nonlinear Phenomena* 22, 120–149 (1986).
- [98] Chris G. Langton. "Computation at the edge of chaos: Phase transitions and emergent computation". *Physica D: Nonlinear Phenomena* 42, 12–37 (1990).
- [99] Wentian Li, Norman H. Packard, and Christopher G. Langton. "Transition Phenomena In Cellular Automata Rule Space". *Physica D: Nonlinear Phenomena* 45, 77–94 (1990).
- [100] Philippe M. Binder. "A phase diagram for elementary cellular automata". *Complex Systems* 7, 241–247 (1993).
- [101] Robert H Gilman. "Classes of linear automata". *Ergodic theory and dynamical systems* 7, 105–118 (1987).
- [102] Petr Kurka. "Languages, equicontinuity and attractors in cellular automata". *Ergodic Theory and Dynamical Systems* 17, 417–433 (1997).
- [103] Petr Kurka. "Topological dynamics of cellular automata". *Codes, Systems, and Graphical Models*, 447–485 (2001).
- [104] Martin Schüle and Ruedi Stoop. "A full computation-relevant topological dynamics classification of elementary cellular automata". *Chaos: An Interdisciplinary Journal of Nonlinear Science* 22, 043143 (2012). issn: 1089-7682. url: <http://dx.doi.org/10.1063/1.4771662>.
- [105] Wentian Li. "Power Spectra of Regular Languages and Cellular Automata". *Complex Systems* 1, 107–130 (1987).
- [106] Howard A. Gutowitz, Jonathan D. Victor, and Bruce W. Knight. "Local structure theory for cellular automata". *Physica D: Nonlinear Phenomena* 28, 18–48 (1987).
- [107] Howard A. Gutowitz. "A hierarchical classification of cellular automata". *Physica D: Nonlinear Phenomena* 45, 136–156 (1990).
- [108] G Braga et al. "Pattern growth in elementary cellular automata". *Theoretical Computer Science* 145, 1–26 (1995).
- [109] Andrew Wuensche. "Classifying cellular automata automatically: Finding gliders, filtering, and relating space-time patterns, attractor basins, and the Z parameter". *Complexity* 4, 47–66 (1999).
- [110] Hector Zenil. "Compression-based investigation of the dynamical properties of cellular automata and other systems". *Complex Systems* 19, 1–28 (2010).
- [111] Genaro J. Martinez. "A note on elementary cellular automata classification". *Journal of Cellular Automata* 8, 233–259 (2013).

- [112] Stephen Wolfram. *Random sequence generation by cellular automata*. Vol. 7. 2. 1986, pp. 123–169.
- [113] *Random Number Generation*. url: <https://reference.wolfram.com/language/tutorial/RandomNumberGeneration> (visited on 02/08/2021).
- [114] Matthew Cook. “Universality in Elementary Cellular Automata”. *Complex Systems* 15, 1–40 (2004).
- [115] James E. Hanson and James P. Crutchfield. “Computational mechanics of cellular automata: An example”. *Physica D: Nonlinear Phenomena* 103, 169–189 (1997).
- [116] Bruno Martin. “A group interpretation of particles generated by one-dimensional cellular automaton, Wolfram’s rule 54”. *International Journal of Modern Physics C* 11, 101–123 (2000).
- [117] Genaro J. Martínez, Andrew Adamatzky, and Harold V. McIntosh. “Complete characterization of structure of rule 54”. *Complex Systems* 23, 259–293 (2014).
- [118] Peter Grassberger. “Chaos and Diffusion in Deterministic Cellular Automata”. *Physica D*, 52–58 (1984).
- [119] Kari Eloranta and Esa Nummelin. “The kink of cellular automaton rule 18 performs a random walk”. *Journal of Statistical Physics* 69, 1131–1136 (1992).
- [120] Peter Grassberger. “Long-range effects in an elementary cellular automaton”. *Journal of Statistical Physics* 45, 27–39 (1986).
- [121] Genaro Juárez Martínez, Andrew Adamatzky, and Harold V. McIntosh. “Phenomenology of glider collisions in cellular automaton Rule 54 and associated logical gates”. *Chaos, Solitons and Fractals* 28, 100–111 (2006).
- [122] Daisuke Uragami and Yukio Pegio Gunji. “Universal emergence of $1/f$ noise in asynchronously tuned elementary cellular automata”. *Complex Systems* 27, 399–414 (2018).
- [123] Genaro J Martínez et al. “How to make dull cellular automata complex by adding memory: Rule 126 case study”. *Complexity* 15, 34–49 (2010).
- [124] Prakash Murali et al. “Software Mitigation of Crosstalk on Noisy Intermediate-Scale Quantum Computers”. *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems* (2020).