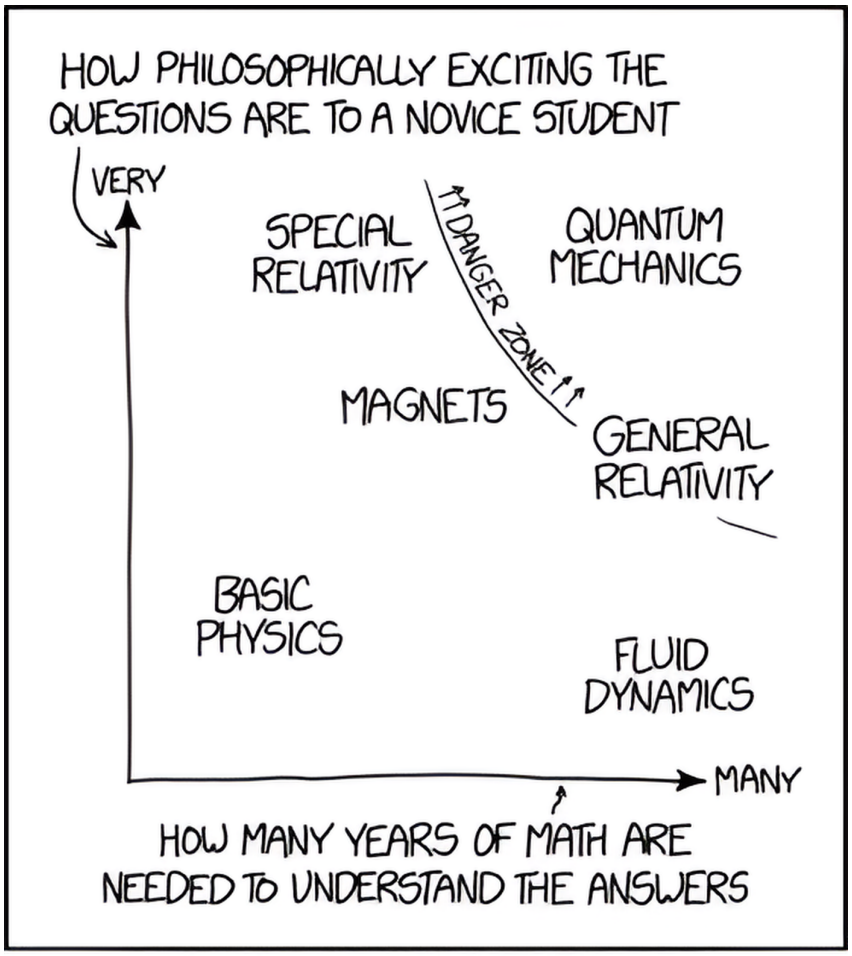
Quantum Cellular Automata

NIUS Physics (Batch 20)
Mid-Term Report
Camp: 20.2

Piyush Kumar Singh* IISER Kolkata

Sambuddha Sanyal† IISER Tirupati

January 23, 2024



WHY SO MANY PEOPLE HAVE WEIRD IDEAS ABOUT QUANTUM MECHANICS

*pks22ms027@iiserkol.ac.in
†sambuddha.sanyal@iisertirupati.ac.in

Acknowledgement

This work was supported by the National Initiative on Undergraduate Science (NIUS) undertaken by the Homi Bhabha Centre for Science Education, Tata Institute of Fundamental Research (HBCSE-TIFR), Mumbai, India. We acknowledge the support of the Department of Atomic Energy, Govt. Of India, under Project Identification No. RTI4001.

Contents

1 Introduction

Cellular automata are discrete mathematical models used to simulate complex systems. John von Neumann first introduced the concept [5] in 1966. In 1970, an article written by Martin Gardner [4] introduces us to a compelling use case of this abstract concept, "The Game of Life," invented by J. H. Conway.

In a conference in 1982, R. P. Feynman expressed his view (or dream) of using 'Quantum Physics' in computers [1] to simulate complex physical systems using the idea of density matrix (proposed by Neumann in 1955 [6]); building on his idea Feynman introduced the world to a weird collaboration named "Quantum Cellular Automata" (QCA) in an article published in 1986 [2].

And there are multiple reasons why we should care about QCAs. First, this is part of a broad field of 'Quantum Information Processing,' any development in QCA may help us understand how we can harness quantum properties for computation. Second, the classical systems exhibit some new emergent behaviors, so we want to know if we can also have these emergent behaviors in QCAs.

Also, in the later part of the discussion, we will see the application of these 'discrete' quantum systems (in the form of "Random Quantum Circuits"[3]), explaining some of the various emergent properties like:

(a) Periodic Fidelity,
(b) Information spread (encoded as *Operator Spread*).

§1.1 Cellular Automata

A cellular automaton (CA) is a group of "coloured" cells on a predetermined-shaped grid that follows a set of rules depending on nearby cell states to evolve over many discrete time steps. After that, the rules are applied repeatedly for as many time steps as needed.

It turned out that reaction-diffusion systems, biological pattern development, fluid flows, and traffic models are only a few real-world uses for CAs. Lattice gas cellular automata are a specific type of CA used to simulate fluid dynamics. The Navier-Stokes equation can be obtained by selecting the appropriate model. Lattice Boltzmann models have taken their place, using continuous functions at the lattice locations rather than discrete variables. More aspirationally, CAs have been proposed as discrete physics models with many desirable characteristics, such as dynamical homogeneity and localization. But, while this is a fascinating idea, physics is ultimately quantum, and CAs cannot adequately represent, e.g., Bell inequality violation, which occurs from quantum entanglement.

Before giving a formal definition, let's look at an example:

Example 1 (Rule Number 110)

Consider a one-dimensional array of bits (*i.e.* cells having two states, namely '0' or '1'). The update rule of a bit is dependent on the initial state of that bit as well as the state of its neighbouring bits. The rule can be summarized as follows:

Initial state of a bit and its neighbours	111	110	101	100	011	010	001	000
New state of middle bit	0	1	1	0	1	1	1	0

Table 1: Update rule for a *one*-dimensional CA

We named this update rule as `Rule 110` because if you treat the last row of the table as a binary number, it converts to 110 in the decimal system. This CA can simulate a Turing machine efficiently (with only polynomial overhead).

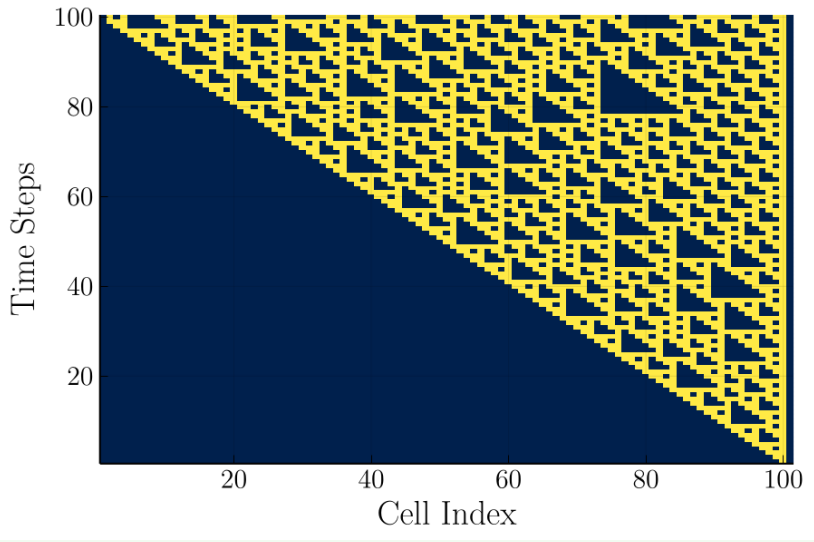


Figure 1: An example of a CA evolution for the rule 110 CA, with time going up. Bits with value 1 are represented by the yellow squares, while blue squares represent 0.

Definition 2 (Cellular Automata):
A Cellular Automaton is a 4-tuple $(L, \Sigma, \mathcal{N}, f)$ consisting of a d-dimensional lattice of cells indexed by integers, $L = \mathbb{Z}^d$, a finite set Σ of cell states, a finite neighbourhood scheme $\mathcal{N} \subseteq \mathbb{Z}^d$, and a local transition function $f : \Sigma^{\mathcal{N}} \to \Sigma$.

To better understand this local transition function f, consider a cell $x \in L$. This function takes the state of the neighbours of x as the argument, which is indexed by the set $\mathcal{N}$ at the current time $t \in \mathbb{Z}$ to spit out the state of cell x at time $t + 1$.
This observation points us towards two important properties of cellular automata that will help us understand why CAs are very helpful in simulating certain systems:

- cellular automata are `space-homogeneous` *i.e.* the local transition function performs the same function at each cell.
- Also, cellular automata are `time-homogeneous` *i.e.* the local transition function does not depend on the current time t.[1]

We can define the current state of the lattice (or CA) as a *configuration* $C \in \Sigma^L$, which has the information about the state of each cell at time t. Now, using this idea, we can define a '*global*' transition function $F : \Sigma^L \to \Sigma^L$, which acts on the entire latter rather than on individual cells and spits out another configuration C' for the lattice at time $t + 1$.

Remark 3 (Revisiting Rule 110). Observe the example 1; we can fit this scenario in the definition. In this case $L = \mathbb{Z}$, $\Sigma = \{0, 1\}$, for i^{th} cell $\mathcal{N} := \{i - 1,\ i,\ i + 1\}$ and f is defined according to the update rule specified in table 1.

[1] We will review these properties in detail in sections 1.1.2 and 1.1.3

§1.1.1 Characteristics of Cellular Automata

In the definition and example, we have used a few terms; now we will define them (and these will be the main characteristics of cellular automata):

Discrete Space-Time: In our discussion, we have defined the lattice (L) to be d-dimensional integer space, which makes our area of interest a discrete space. Our evolution to the next state is also discrete since we evolve our states from $t \to t+1$ (or predefined time steps).

Finite State Set: The set Σ is finite *i.e.* every cell has only finitely many states of being in. Generally, in classical systems, we have state set $\Sigma = \{0, 1\}$, also called 'binary cellular automata.'

Cell/Cell-state: Every space-position vector $i \in L = \mathbb{Z}^d$ identifies a cell which contains a cell-state $c_i \in \Sigma$.

Configuration: A configuration C is the concatination of all cell-states over the entire lattice L at a given time t which means that every configuration is an element of set Σ^L.

Local Transition Function: We know time is also discrete in our system (with a predefined time step). We use a local transition function f that defines the state of every cell after each time evolution. We refer to this function as a local function, as it only takes the states of the 'neighbouring' cells as an argument.

Neighbourhood Scheme: As we have defined, the outcome of our local function at any cell i depends on the set of cell-states of neighbouring cells. This set is defined by the neighbourhood scheme $\mathcal{N}$ on cellular automata.

Global Function: For a given lattice L (with a defined neighbourhood scheme), the local transition function f imposes a global transition function F on the set of all possible configurations. This global function determines the *space/time* behavior of the cellular automaton on an initial configuration.

§1.1.2 Locality

In physics, locality implies we are talking about the Principle of Locality, which states that an object is influenced directly only by its immediate surroundings. If a theory that includes this principle is said to be a "local theory." This idea of locality evolved from the classical field theory; building upon this idea, if an event at a point is cause for an effect at another point, a wave or a particle must travel through space-time between the two points, carrying the influence.

Einstein postulated that causal influence couldn't be transmitted between two points faster than the speed of light c, in the Special Theory of Relativity, which means that if two points are spacelike separated, they can't influence each other.

In the context of cellular automata, we know that the state of a cell is only influenced by the states of finite neighbours (decided by neighbourhood scheme). Since the state of any cell is influenced by its surroundings, we can say that the cellular automata theory is local.

§1.1.3 Universality

§1.2 Quantum Cellular Automata

R. P. Feynman introduced the idea of Quantum Cellular Automata with a vision that it can be used to simulate 'Quantum Physics' as we don't have a classical analog of many quantum phenomena. In [7], a group of scientists has proven that QCAs are an alternative paradigm of quantum computers. They have shown QCAs to be universal, implying they could efficiently simulate a quantum Turing machine.

This demonstration showed that QCAs are very important for quantum computers, so many have tried to formulate a theory. But there was an enormous challenge in front of the scientists: after some local function has updated one cell, we will lose the information about the state of the cell at time t as we require it for updating other cells. So somehow, we have to clone the state of this cell to some other cell, but by the no-cloning theorem, it is simply impossible.

2 Random Quantum Circuits

A Notations

§A.1 Hello

Bibliography

[1] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6-7 (June 1982), pp. 467–488. ISSN: 1572-9575. DOI: 10.1007/bf02650179. URL: http://dx.doi.org/10.1007/BF02650179.

[2] Richard P. Feynman. "Quantum mechanical computers". In: *Foundations of Physics* 16.6 (June 1986), pp. 507–531. ISSN: 1572-9516. DOI: 10.1007/bf01886518. URL: http://dx.doi.org/10.1007/BF01886518.

[3] Matthew P.A. Fisher et al. "Random Quantum Circuits". In: *Annual Review of Condensed Matter Physics* 14.1 (Mar. 2023), pp. 335–379. ISSN: 1947-5462. DOI: 10.1146/annurev-conmatphys-031720-030658. arXiv: 2207.14280v1 [quant-ph]. URL: http://dx.doi.org/10.1146/annurev-conmatphys-031720-030658.

[4] Martin Gardner. "Mathematical Games". In: *Scientific American* 223.4 (Oct. 1970), pp. 120–123. ISSN: 0036-8733. DOI: 10.1038/scientificamerican1070-120. URL: http://dx.doi.org/10.1038/scientificamerican1070-120.

[5] John von Neumann. "Theory of Self-Reproducing Automata". In: (1966). URL: https://cba.mit.edu/events/03.11.ASE/docs/VonNeumann.pdf.

[6] John Von Neumann. *Mathematical foundations of quantum mechanics*. en. Ed. by Nicholas A Wheeler. Princeton, NJ: Princeton University Press, Feb. 2018.

[7] J. Watrous. "On one-dimensional quantum cellular automata". In: *Proceedings of IEEE 36th Annual Foundations of Computer Science*. 1995, pp. 528–537. DOI: 10.1109/SFCS.1995.492583.