Glitches in the Matrix? From Dead Reckoning for 'Believable' Peer-to-Peer Gaming to a Many-Interacting-Simulations Explanation of Quantum Mechanics

Marcus Arvan
The University of Tampa
marvan@ut.edu

Abstract: This paper outlines how a potential new interpretation of quantum mechanics—*the Many-Interacting-Simulations (MIS) interpretation*—may explain why our world has many of the bizarre quantum features it does. §1 provides an overview of what has been termed the Peer-to-Peer (P2P) Simulation Hypothesis, showing how it explains general features of quantum phenomena. §2 then examines *dead reckoning*, a framework utilized to ensure 'believable' simulated worlds for P2P networked games. §3 details how dead reckoning in P2P games reproduces—and hence, computationally explains—properties broadly analogous to (§3.1) quantum mechanical standing waves, (§3.2) Gaussian wave packets, (§3.3) the double-slit experiment, and (§3.4) the 'Wigner's friend' thought experiment. §4 then details how, if this explanation is correct, then traditional interpretations of quantum mechanics each contain elements of truth. §5 concludes that quantum phenomena may be the 'glitches in the Matrix' that many have suggested would be evidence that we live in a computer simulation.

Key words: dead reckoning, double-slit, quantum mechanics, simulation hypothesis

> At the basis of the whole modern view of the world lies the illusion that the so-called laws of nature are the explanations of natural phenomena.
>
> - Ludwig Wittgenstein[1]

Quantum mechanics is an exquisitely well-verified mathematical theory of microphysics. There are, in turn, many competing theories of how to interpret its physical significance, ranging from the Copenhagen Interpretation[2], to Bohmian Mechanics[3], to the Many Worlds Interpretation[4], and beyond.[5] However, there are well-known objections to every leading interpretation[6], and importantly, each interpretation merely purports to explain what

*Many thanks to Ben Hoyt (Founder & CEO, *47 Games*) for bringing dead reckoning programming strategies to his attention, and for helpful discussion. I also thank audiences at *Munich Center for Mathematical Philosophy*, Ludwig Maximilian University of Munich and *The Many-Worlds Interpretation of Quantum Mechanics: Current Status and Relation to Other Interpretations* Research Workshop of the Israeli Science Foundation, Center for Quantum Science and Technology, Tel Aviv University, 2022.

quantum mechanics is, not *why* it is.[7] For example, the Copenhagen interpretation holds, very roughly, that every particle exists in many different states at once (in a 'coherent superposition'), and that measurements cause the superposition to 'collapse', such that the particle will be observed to be in a single state following the probability amplitudes described by the quantum wave-function.[8] Yet, even assuming this to be correct, the Copenhagen Interpretation does not explain why our world is this way—that is, *why* our world has particles that exist in superpositions only to collapse to particular eigenstates when measured. In contrast, the Everett/Many Worlds Interpretation holds that particles exist in many 'worlds' (or parallel quantum universes)[9], and that through a process of quantum decoherence, the surrounding conditions in each universe select which particular state (or eigenvalue) a particle will have in that universe following the quantum wave-function.[10] Yet, once again, even assuming this to be correct, the Many Worlds Interpretation does not explain why the world is this way—that is, *why* our world is part of a much larger multiverse that constantly decoheres into new divergent universes following the probabilistic mathematics of the wave-function. Finally, Bohmian Mechanics holds that there is just one universe (contra the Many-Worlds Interpretation), that particles always exist in determinate states rather than superpositions (contra the Copenhagen interpretation), and that particles are in deterministically guided by pilot waves with 'hidden variables' that only allow observers to predict particle states probabilistically.[11] Yet, as with the other interpretations, Bohmian Mechanics does not explain why our world is this way—that is, *why* our world (supposedly) has pilot waves with 'hidden variables' that govern particle behavior.

This paper argues that a potential new interpretation of quantum mechanics—based on state-of-the art approaches to programming 'believable' environments in certain types of online videogames—may explain why our world has many of the bizarre quantum mechanical features it does. Section 1 briefly summarizes what has been termed the Peer-to-Peer (P2P) Simulation Hypothesis[12], showing how this hypothesis has been argued to provide a unified conceptual explanation of a variety of quantum phenomena—ranging from quantum superposition to wave-particle duality, to wave-function collapse—but also how it has until now lacked a clear mechanism for explaining other details of quantum mechanics. Section 2 then introduces *dead reckoning*, a mathematical framework commonly used in real-world navigation to calculate the current and likely future position(s) of an object using its previously determined position, velocity, and direction[13], and which is also used in computer networking for online games that utilize P2P architecture.[14] Section 3 then outlines how state-of-the-art approaches to dead reckoning in P2P games to ensure a 'believable' gaming environment—approaches utilizing velocity and path-blending, as well as cubic Bézier splines, centripetal Catmull–Rom splines, and Hermite curves[15]—reproduce and hence explain in computational terms properties *broadly analogous to* our world's quantum wave-function (§3.1), the super- and sub-structures of Gaussian wave packets (§3.2), the double-slit experiment (§3.3), and (§3.4) why consciousness appears to play a privileged and paradoxical role in measuring quantum states (qua 'Wigner's friend' thought experiment[16]). Section 4 then briefly details how, if this explanation of quantum phenomena is correct, then different traditional interpretations of quantum mechanics each contain elements of truth. Section 5 concludes that dead reckoning in P2P gaming may explain why our Universe is quantum-mechanical, something that no other interpretation of quantum mechanics even purports to achieve—and hence, that quantum phenomena may be the very kind of 'glitches in the Matrix' that many have suggested would be concrete evidence that we live in a computer simulation.[17]

Before proceeding, several important caveats are necessary. First and foremost, this paper is merely intended to be a *high-level conceptual exploration* of how the functional properties of certain types of computer simulations reproduce—and thus may provide a kind of unified computational explanation of—certain quantum features of our world that are currently taken to be fundamental in science. As a high-level conceptual exploration, this paper does *not* aim to capture or explain quantum mechanics in total, or even a high fraction of quantum mechanics' rich phenomenology, qua mathematical agreement with the actual kinematics of quantum mechanics. Obviously, any adequate explanatory theory of quantum phenomena must ultimately achieve this. The devil, as they say, is in the details—and any adequate theory of the quantum world must be in calculative agreement with physical equations. Why, if this paper does not even purport to achieve this kind of agreement, is this paper worth introducing into philosophical and scientific discussion?[18] The answer is that conceptual explorations of new ways to explain phenomena can be necessary and legitimate precursors to more detailed, empirically adequate theoretical development—serving, indeed, to warrant such development. For example, when proposing evolution by natural selection as a unified explanation of the origin and development of species, Charles Darwin lacked anything remotely close to an empirically adequate theory of genetics (or indeed, epigenetics). An empirically adequate understanding of genetic variation, heritability, and forces of natural selection has taken centuries—and indeed, is still very much in progress. But this is precisely what Darwin's conceptual exploration of evolution by natural selection—and his broad-brush demonstration of how the theory plausibly explained observed changes in animal species in various field-work—was good for. Darwin demonstrated, at a broad conceptual level, that evolution by natural selection was a promising enough potential explanation of the origin and development of species to justify further, more detailed examination by other scientists. While this paper is of course nothing on the order of *On the Origin of Species*—a vast work of nearly unparalleled depth and breadth in the history of human inquiry—its basic aim is similar: to sketch how a *new kind of explanation* (in this case, a computational explanation of physics utilizing computer networking) is at least promising enough at a conceptual level to warrant further investigation by mathematicians, physicists, and computer scientists. Finally, although this paper does not demonstrate detailed calculative agreement between its model and physical equations, it is worth bearing in mind that some fundamental components of the model—its understanding of quantum phenomena in terms of many-interacting simulations—can be related to a preexisting interpretation of quantum mechanics that *has* been developed at a more detailed level: the many-interacting worlds interpretation.[19] If this paper is broadly successful in its aims, it may provide a deeper account of how to best understand what the "many-interacting worlds" in this interpretation *are* (namely, simulated worlds), and how computer networking may provide a deeper explanation of how any why these "multiple worlds" interact as they do.

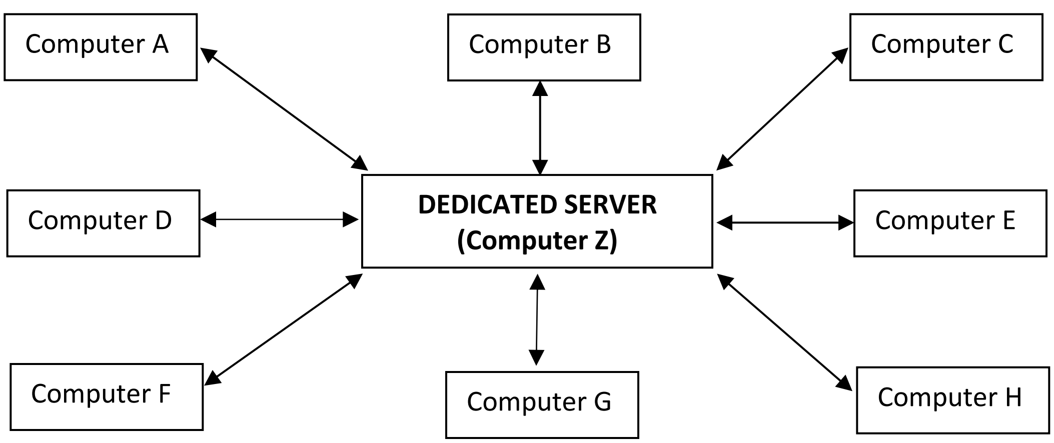
1. The P2P Simulation Hypothesis: A Brief Overview

The idea that we live in a simulation is highly speculative. However, it appears to be taken seriously by an increasing number of philosophers[20], physicists[21], computer scientists[22], and programmers.[23] Further, several theorists have argued that the hypothesis better explains quantum phenomena than standard interpretations using ordinary physical theories. For example, Emery contends that the 'brain-in-a-vat hypothesis' provides an elegant explanation of the quantum EPRB correlation—the experimental finding that two particles separated by a distance are always correlated despite proofs that there can be no causal signal or common physical cause for the correlations.[24] More ambitiously, Arvan contends that a novel version of the simulation hypothesis—the Peer-to-Peer (P2P) Simulation Hypothesis—explains a variety of quantum phenomena, ranging from quantum superposition to wave-particle duality and quantum entanglement.[25] As this hypothesis will be central to our investigation, let us briefly summarize its main components.

Computer networking comes in two main forms. First, there are centralized networks (Figure 1), where there is a single computer or 'dedicated server' that other computers on the network access to retrieve information.

Figure 1.

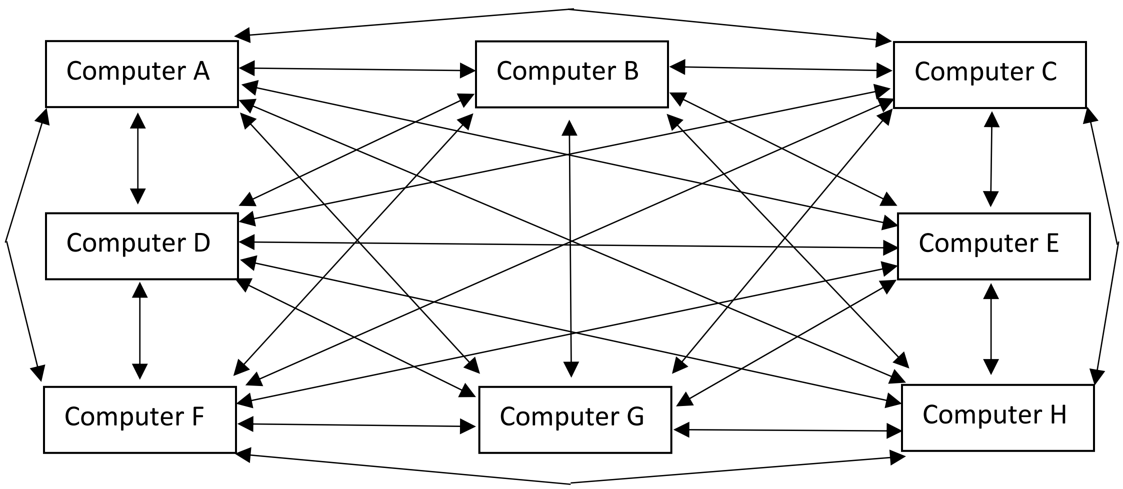
Centralized 'Dedicated Server' Computer Network



Second, there are distributed networks, otherwise known as peer-to-peer P2P client networks (Figure 2). In P2P networks, there is no dedicated server. Instead, each computer on the network serves as its own 'node', which other computers on the network communicate with directly to exchange information.

Figure 2.

Distributed 'Peer-to-Peer' (P2P) Computer Network



Next, let us compare the functional properties of both types of networking as they apply to *simulated realities*. Many online videogames—ranging from Minecraft to Ragnarok—utilize dedicated servers.[26] The main advantages of dedicated servers are that the 'host' server 'transmits data to all the players and keeps the game in sync.'[27] This means that,

The host handles:

- Every player's location in the game.
- The activity of all the players.
- Player interactions.
- Each character's loadout.
- What direction all players are facing.
- The physics happening around each player.
- Score and rules.[28]

In other words, in a dedicated server game, every player and object in the simulated world has a *single determinate location* and *other determinate 'physical properties'* (such as direction of movement, velocity, acceleration, etc.). The main downsides of dedicated servers in gaming are that, 'The host machine requires more memory, bandwidth, and processing power the more players log in and play the game', and that, 'If the host player has a weak computer or a bad internet connection, the game will suffer from lag, bugs, and crashes.'[29] In other words, it is extremely difficult, the more players there are, for dedicated servers to achieve a seamless and stable gaming experience without noticeable glitches—such as players 'lagging' (or appearing to shift positions randomly in the game)—and an increasingly high probability of the simulation crashing.

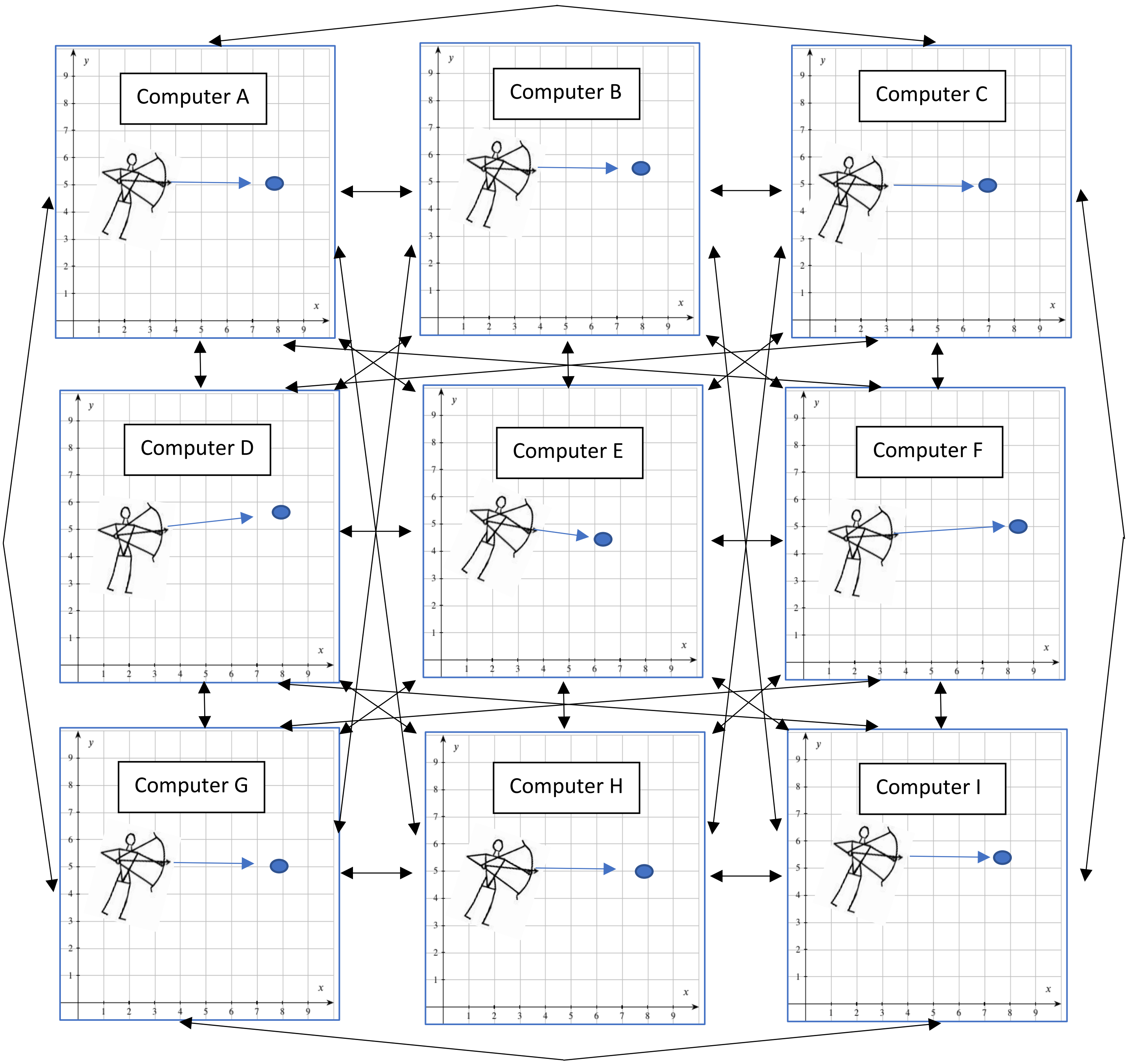
P2P games resolve these problems by taking an altogether different approach. Instead of having a 'host' computer that records where all objects and players 'objectively' are in the game, P2P simulations are comprised by *each* computer on the network running their own game simulation—or a small part of it[30]—in parallel with all other computers on the network. Because P2P networks distribute computing to every computer on the network, they are especially suitable for *massive multiplayer games* with large numbers of players that would overwhelm a single dedicated server: a P2P system 'dynamically scales with the number of online players' and 'is more flexible and has a lower deployment cost than centralized games servers.'[31] Importantly, although individual computers on a P2P network can crash, causing the player whose computer crashes to 'die', such node failures in P2P networks are independent, their general frequency is low, and messages can be routed to nearby nodes that are numerically closest to the crashed node.[32] This makes P2P networked games extremely stable, such that 'even when half of the nodes fail simultaneously', it is still 'reasonable to assume that messages eventually reach the correct node', preserving the game experience for all players whose individual computers are still running.[33] While P2P games have downsides—ranging from difficulties in creating P2P architectures with good performance to security issues[34]—the relevant point is that, if we do live in a computer simulation, a P2P network would be strongly favored over a dedicated server. If, for example, we assume that all 8+ billion human beings on Earth today are 'players', a P2P simulation with 8+ billion nodes would be far more stable than a dedicated server. And, of course, there may be far more 'players' than this, if there are for example alien species on other planets, or if nonhuman animals are 'players' as well.

In contrast to dedicated server games, where the 'host' computer again serves as a definitive, 'objective' record of every object and player's location and behavior, P2P games have no such definitive record. Instead, each computer on the network has its own unique representation of where objects and players are in the game are located and how they are behaving. Further, because 'All information that is sent between players may be subject to network latency and packet loss', and because 'it is prohibitively expensive to send information at the same rate that the game updates', individual computers on the network must 'extrapolate from periodic updates', giving rise to *errors* (or different object locations and other properties—such as direction, velocity, etc.—on different computers).[35]

These errors become more pronounced when things move in the game at high speeds.[36] If, for example, we imagine a moving player shooting a projectile in a P2P game, different computers on the network will—due to latency, lag, or packet loss—represent *that player and projectile* in somewhat different locations, moving in subtly different directions, at subtly different speeds (Figure 3). That is, different computers will record objects as being in different *eigenstates*, 'a state of a quantized dynamic system … in which one of the variables defining the state … has a determinate fixed value.'[37]

Figure 3.

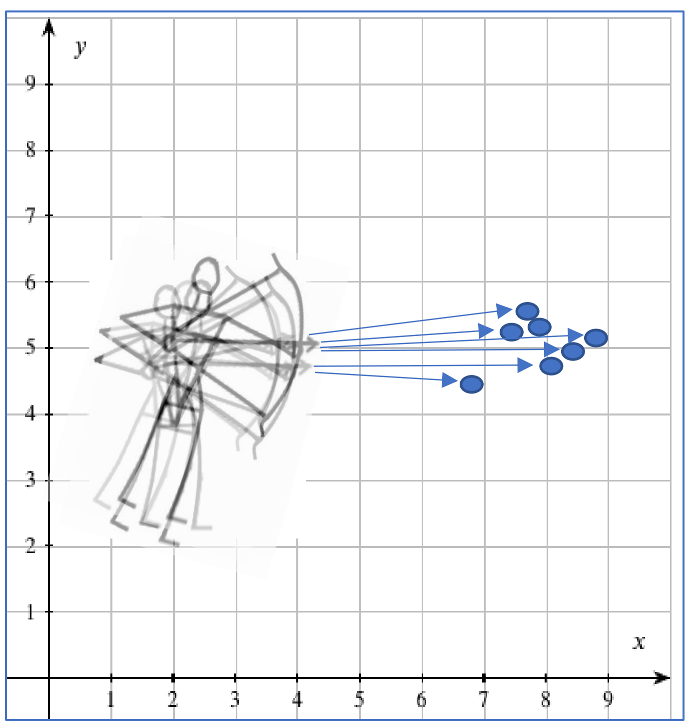
Locations of a 'Single' Player and Projectile on Different Nodes in a P2P Game[38]



It is important to note that in a well-programmed P2P simulation, the kinds of divergences in object placement across nodes illustrated above will normally be too small for players to notice at a 'macroscopic' level in the game. Provided a P2P simulation is suitably programmed to ensure a 'believable' reality for all players—an issue that we will turn to later—the way that different nodes in a network represent an object in slightly different positions will be miniscule, so as be *unnoticeable* to players. This is critical, as we will see, because it is how quantum mechanics works in our world. For now, let us now ask—not from the perspective of any single node, but from the perspective of the P2P simulation as a whole, that is, summing across all of the nodes above—where 'the player' and 'the projectile' are located. What we find is that both are *located in many different places at once* (Figure 4).

Figure 4.

Superpositions of a 'Single' Player and Projectile Summing Across P2P Nodes[39]

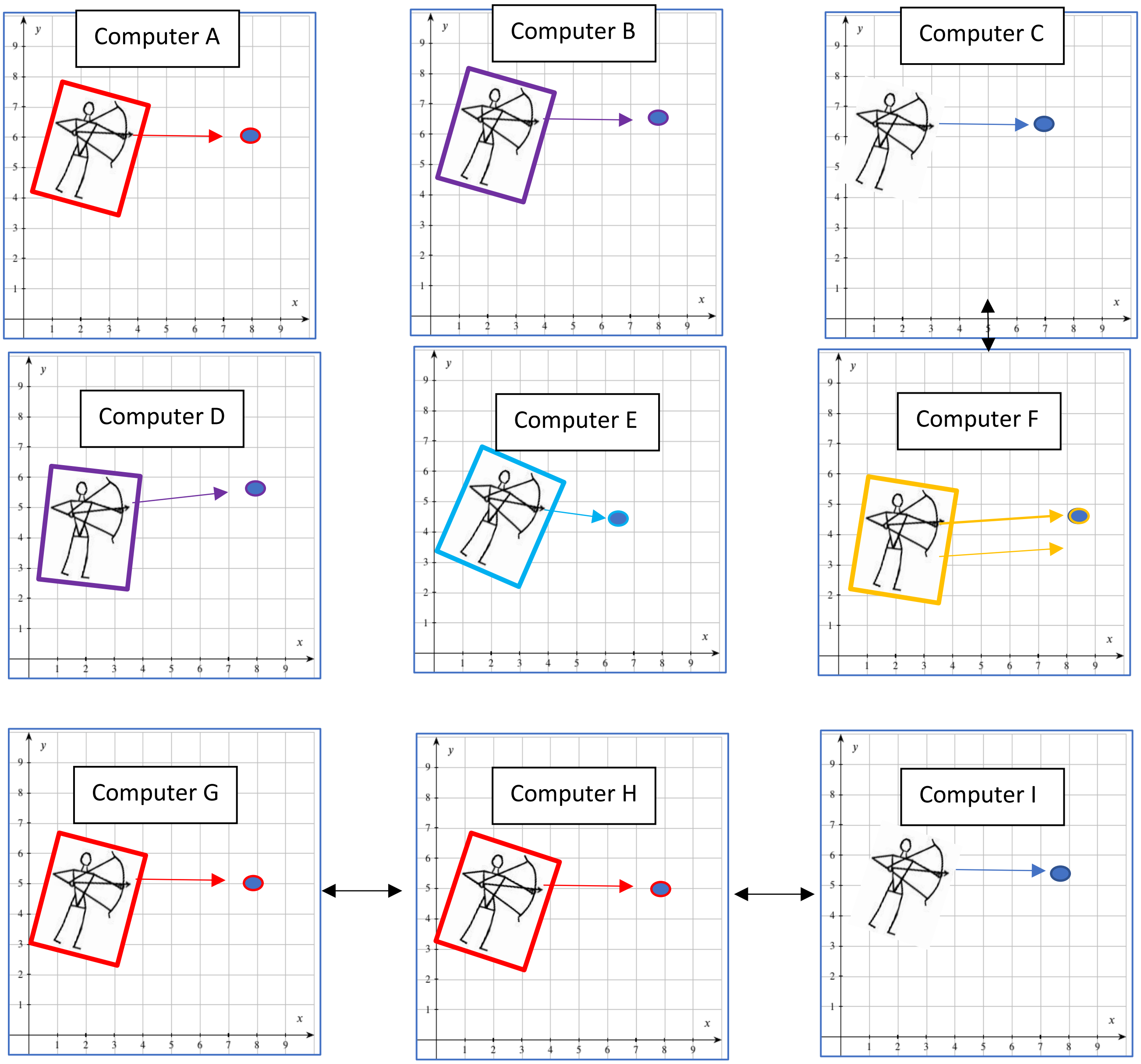


In other words, the player and projectile will both exist in *superpositions* across the network—in different places with different properties on different computers (i.e. slightly different velocities, directions, etc.). At some nodes (Computers A, G, & H), the projectile is represented as moving parallel to the ground/x-axis, and located at (x=8, y=5). However, at other nodes (Computers B & D), the projectile is instead represented as being slightly above this location, at (x=8, y=5.5). Further, at one of these nodes (Computers B), this is because *the player* is represented as firing the projectile from a slightly higher elevation, whereas at the other node (Computers D) the player is represented as firing the projectile at a higher *angle* relative to the ground/x-axis. So, from the standpoint of the P2P simulation as a whole, neither the projectile nor the player *ever* exists in a single determinate state (viz. location, movement, etc.). Yet, the projectile and player have determinate properties at every *individual* node. That is, objects are never represented as being at multiple places on any computer, but summing across nodes, they are always in *multiple, superimposed states* 'at once.'

These facts can be understood as constituting an evolving *probability distribution*. We can see how by focusing on the distribution across nodes of where the player and projectile are each located in Figure 5. Notice, first, that the projectile is represented as being at (x=8, y=5) on three separate nodes (Computers A, G, & H). On all three nodes, the *player* is also represented as being in roughly the same location in the game, their head being located just below and to the left of (x=2, y=6). In contrast, there is only *one* node (Computer F) where the projectile is represented as being at (x=8.5, y=5), *two* nodes (Computers B and D) where the projectile is represented as located at (x=8, y=5.5), etc. (Figure 5).

Figure 5.

Distribution of Projectile and Player Location in a P2P Game at *t*



Red = projectile represented at location (x=8, y=5)

Purple = projectile represented at location (x=8, y=5.5)

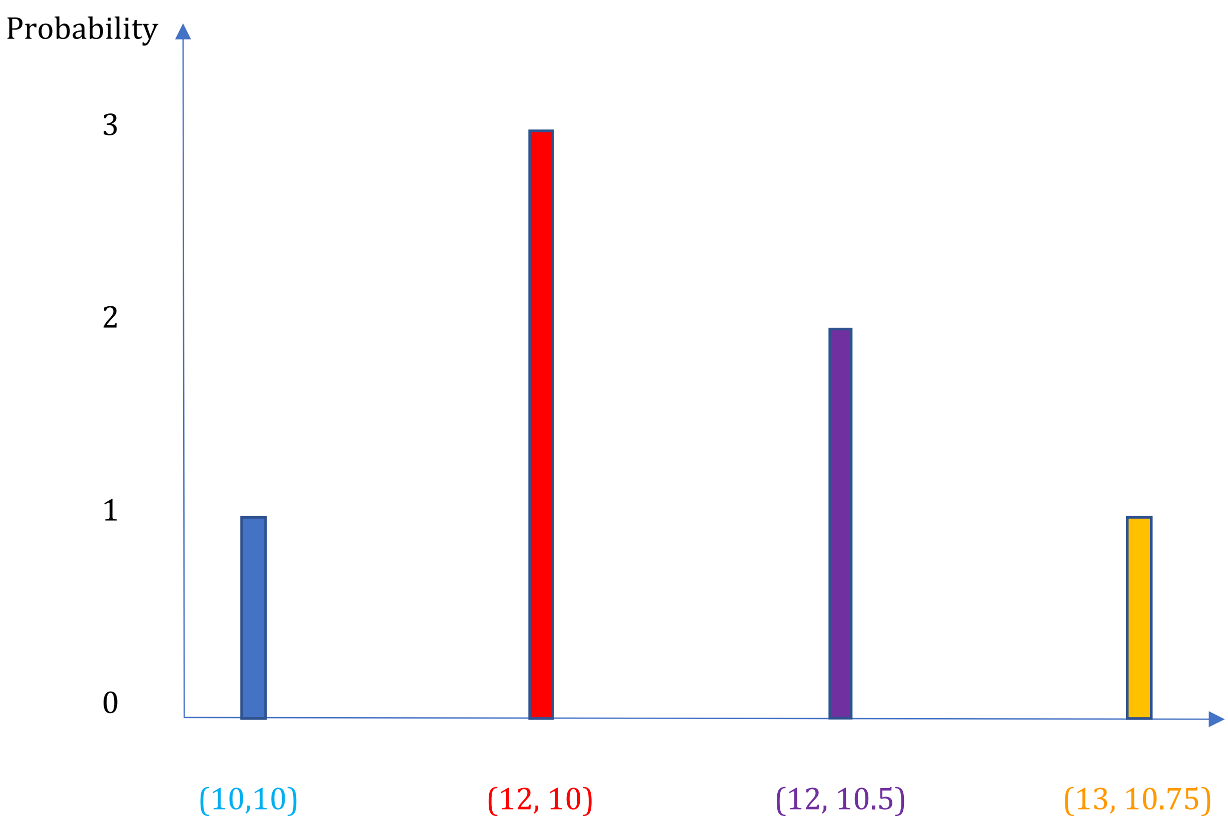
Orange = projectile represented at location (x=8.5, y=4.5)

Blue = projectile represented at location (x=6.5, y=4.5)

The point then is this. Every individual computer in a P2P game will represent any object—such as a projectile—as being located at a particular point in the game, like a *point-particle*. Yet, when we sum across nodes, that object will have an *amplitude* corresponding to how many computers on the network represent it being at that position. So, at time *t*, we may say that location (8, 5) has an *amplitude of 3*, where by this we simply mean that just prior to *t*, a given player in the game would have a *higher probability* of measuring the projectile as being at that location than they would at (8, 5.5), which in turn—insofar as its amplitude is 2—a player would have a higher probability of measuring the projectile to be at than at (8.5, 4.5). We can then plot this probability histogram (Figure 6).

Figure 6.

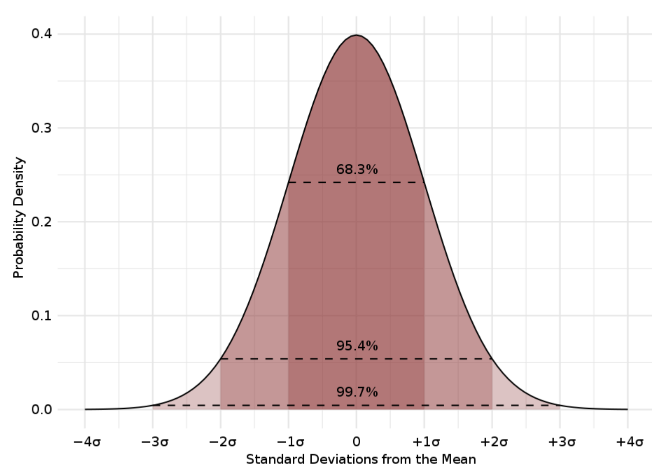
Probability Distribution of Projectile Location in P2P Simulation at *t*



Notice, next, that this probability function will *change over time* as the projectile moves and the different nodes in a P2P simulation communicate with each other to update where they represent the projectile as being. So, for example, although three computers represent the projectile as being at location (8,5) at time *t*, at the very next instant only *one* computer may represent the projectile at that location—thus constituting, from the standpoint of the sum of all nodes, a smaller probability of a player finding the projectile there. This temporal dynamics in a P2P simulation can be shown to have *wave-like features* as follows. Just as in many parts of nature—ranging from the weight and height of different human beings to snowflake sizes, lifetimes of lightbulbs, weights of loaves of bread, and milk production of cows[40]—the measured location of objects by different nodes in a P2P network will, all things being equal, fall approximate *normal distribution*, or 'bell curve' (Figure 7).

Figure 7.

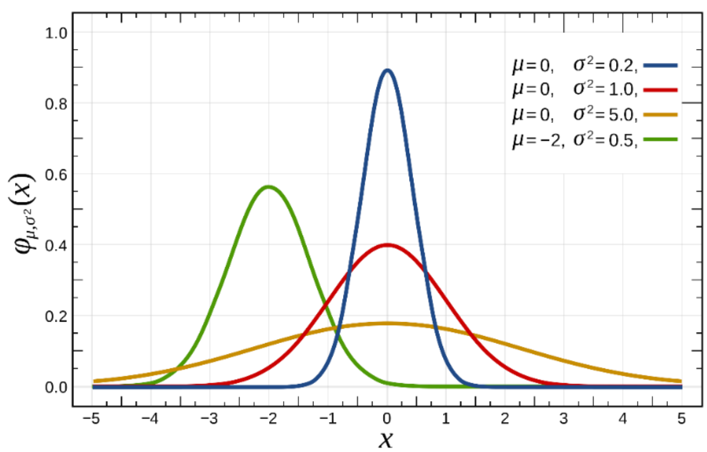
Normal Distribution[41]



This is because *measurement errors* are known to have distributions that are nearly normal.[42] However, P2P networking involves three different sources of measurement errors: (i) latency (the time it takes for information to travel between nodes), (ii) lag (or temporal delays in different computers on a network receiving updates due to network congestion), and (iii) packet loss (i.e., lost packets of data not reaching their destination), measurement errors across P2P nodes constantly occur in P2P networks and should functionally approximate a normal distribution.[43] Lag, in particular, is an emergent feature of a P2P network as a whole, as lag can *increase or decrease* depending on the amount of informational traffic in different parts of the network. As such, moving objects in P2P networks (such as a projectile) should—from the *standpoint of the network as a whole*—constitute *propagating waves* that approximate normal distributions of *fluctuating sizes* (Figure 8).

Figure 8.

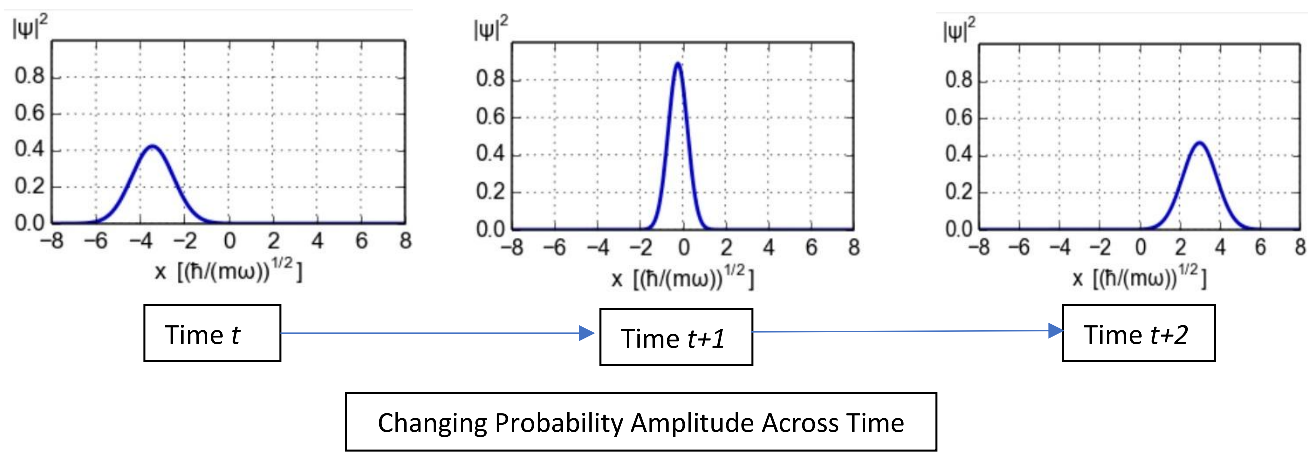
Different Normal Distribution Sizes[44]



As such, although at each instant of time every individual computer on the network will represent the projectile as a 'particle' (with a determinate location), from the standpoint of the network as a whole the projectile's position will approximate the superstructure of a *moving Gaussian wave packet*, which is how quantum information such as electromagnetic waves propagate in our world.[45] For a moving Gaussian wave packet approximates a fluctuating bell curve constituted by a particle's likely location by the quantum wave function (Figure 9)—which as we have seen is precisely how we can understand the informational dynamics of a P2P simulation with different sources of measurement error (we turn to substructures of Gaussian waves shortly).

Figure 9.

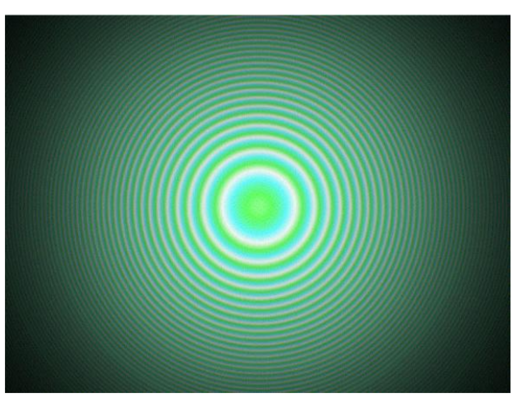
P2P Network Dynamics ≈ Moving Gaussian Wave[46]



Notice, next, that although we have been discussing these matters in two dimensions, online videogames today standardly represent objects three spatial dimensions, in which case the wave-like features of objects in P2P simulations will be *three-dimensional* (Figure 10) while moving through a fourth temporal dimension (e.g., time) in the simulation.

Figure 10.

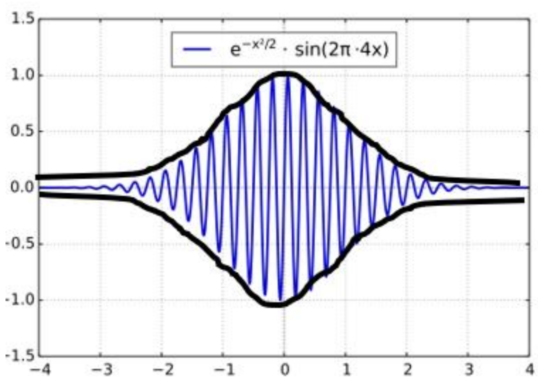
3D P2P Simulation Dynamics ≈ Moving Gaussian Waves in 3 Dimensions[47]



There are, however, features of quantum mechanics that this account does not yet have a clear mechanism to explain. First, Gaussian wave packets have a superstructure (or ‘envelope’) constituted by smaller microstructures (Figure 11).

Figure 11.

Gaussian Wave Packet Superstructure and Microstructure (Ripples)[48]



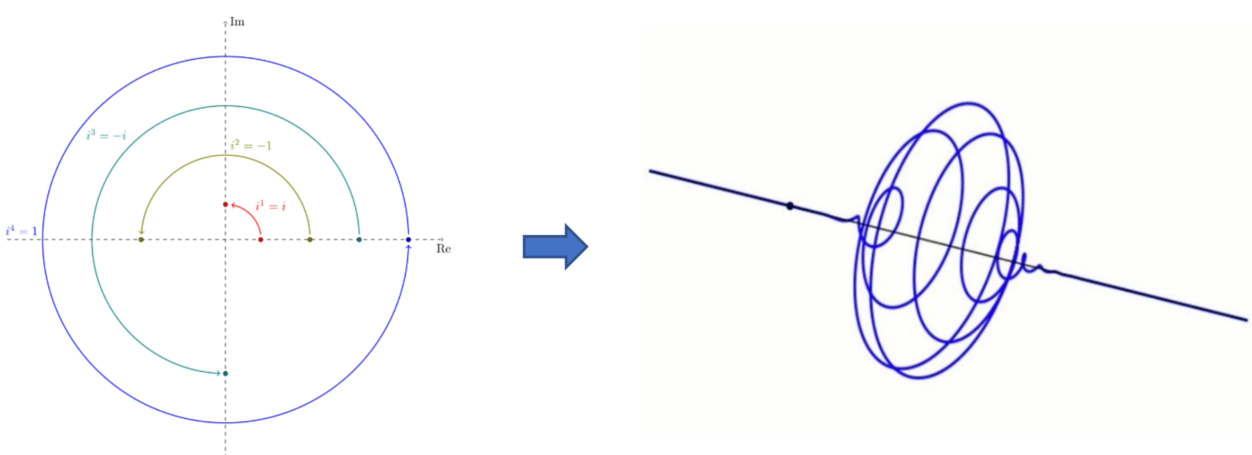
Bold line: “envelope” (superstructure)

Blue line: “ripples” (microstructure)

Second, we need to explain how the microstructure of quantum wave packets are comprised by oscillations involving both *real and imaginary numbers*, where the latter (imaginary numbers) involve *rotations* of quantum waves across spacetime (Figure 12).

Figure 12.

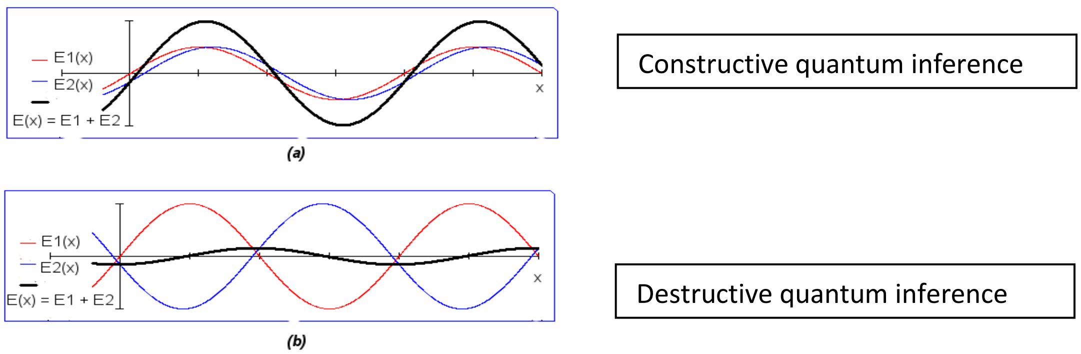
Gaussian Waves w/Imaginary Numbers = Rotating Quantum Wave Packets[49]



Third, we need to explain how, as in quantum mechanics, the Gaussian-wave features of P2P simulations can generate constructive and destructive wave interference[50] (Figure 13).

Figure 13.

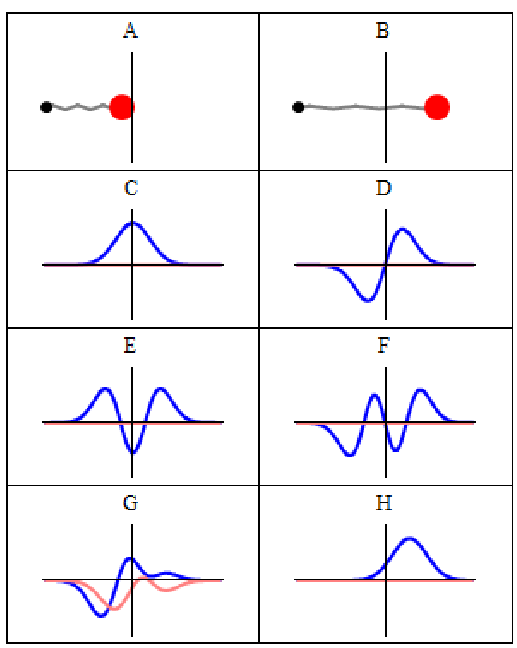
Quantum Interference[51]



Fourth, we need to explain why there are multiple solutions to the Schrödinger equation resulting in different types of standing and propagating waves (Figure 14).

Figure 14.

Classical & Quantum Harmonic Oscillators for a Single Spinless Particle[52]



We will return to these matters momentarily in section 3. The relevant points for now are these: P2P networked games *naturally generate* quantum-like phenomena. All objects in a P2P game are (i) in a constant superposition of different states *across nodes* (or computers) on the network, (ii) have features of particles *at* particular nodes, (iii) features of *waves* as representations change dynamically across nodes due to errors, (iv) corresponding to a *probabilistic* distribution of where any given computer on the network is 'likely' to measure the object to be*.* Notice, further, that P2P games explain why, although objects are in fact superimposed in different locations across nodes, individual users (i.e., game players) never observe that superposition: the superposition is not represented *in* the simulated environment that their computer is presenting to them as an observer. Instead, the superposition only exists summing *across* nodes, which no user experiences in the game.

2. Dead Reckoning and P2P Videogame 'Physics Engines': A Brief Introduction

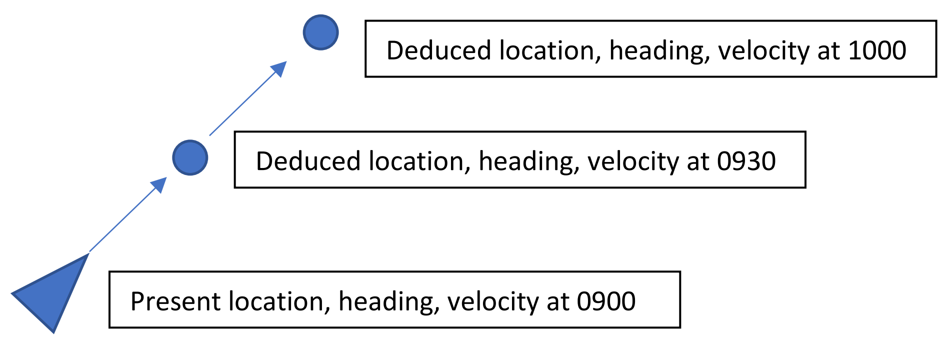
Because P2P game simulations are constituted by many different computers networked together, with each computer rendering a simulated environment for its user in *parallel* to other computers on the network while communicating in real time with other players' computers, programmers must resolve obvious coordination difficulties. If, for example, I shoot a projectile at your character in a P2P game, my computer must send that information (that I have taken a shot in a particular direction, with a particular velocity, at a particular time) to every other computer so that the other players experience the gunshot and subsequent path of the projectile on their end. Yet, this information transfer is constantly reciprocal. If I shoot my gun at your player, who you are moving in some direction to try to avoid getting shot, then your computer must send that information to *my* computer and others on the network as well. Critically, this reciprocal communication between nodes in a P2P network takes time. Many online videogames run at 60Hz (or 60 frames of gameplay per second), some run significantly faster (120 FPS)[53], and different computers on the network need to ensure that different players experience *approximately* the same environment at approximately the same time while sending information packets to each other at a period of under 100ms.[54] For example, if would be disastrous from a gameplay standpoint if one player on a P2P network experienced themselves 'shooting another player' but the other player (and other players on the network) represented their player as avoiding the shot. 'Believable' P2P games need to ensure that all players observe approximately the same things in the simulated environment at approximately the same time, and in ways that involve well-ordered causal sequences across users (rather than contradictory sequences for different players).

This is not easy to do, however. First, programmers must address *latency*, or the speed at which different computers on the network communicate with each other—and it is impractical for computers to send updates to all other computers at the rate that most games run (60Hz). This means that information transfer in a P2P network is always 'behind' what any given player experiences on their device. Second, in addition to latency, computer networks routinely suffer from *lag*, or delays in information reaching their destination, which can result from degradation in internet service provider (ISP) quality, internet connection speed, and geographic distance from other players.[55] Third, computer networking routinely involves *packet loss*, or information never getting to its destination (i.e., to other nodes), which can occur due to network congestion, software bugs, etc.[56]

To address these issues and ensure a 'believable' environment for all players in real time, P2P games standardly utilize a programming solution adapted from real-world navigation known as dead reckoning (or deduced reckoning).[57] In brief, dead reckoning involves predicting where an object is likely to be given its last known location, speed, and direction. So, for example, if we want to know at 9:00am where an airplane is likely to be at some points in the future (e.g., 9:30 and 10:30am), dead reckoning involves deducing this from the object's known speed and direction at 9:00am using linear physics (Figure 15).

Figure 15.

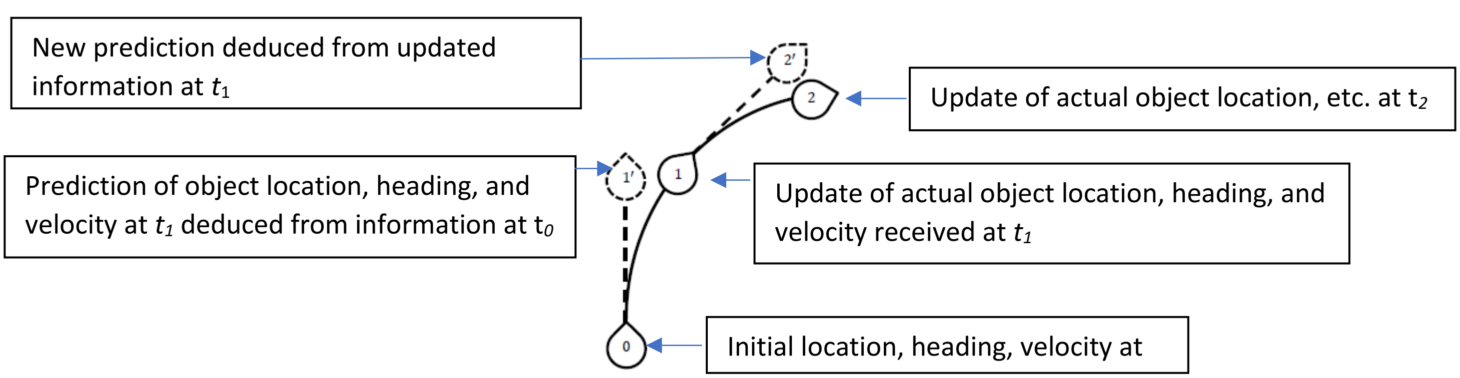
Linear Dead Reckoning Prediction



P2P games do something similar for objects moving in complex directions. Every computer on a P2P network predicts where objects and other players in their environment are likely to be based on *periodic updates* from other computers on the network (Figure 16).

Figure 16.

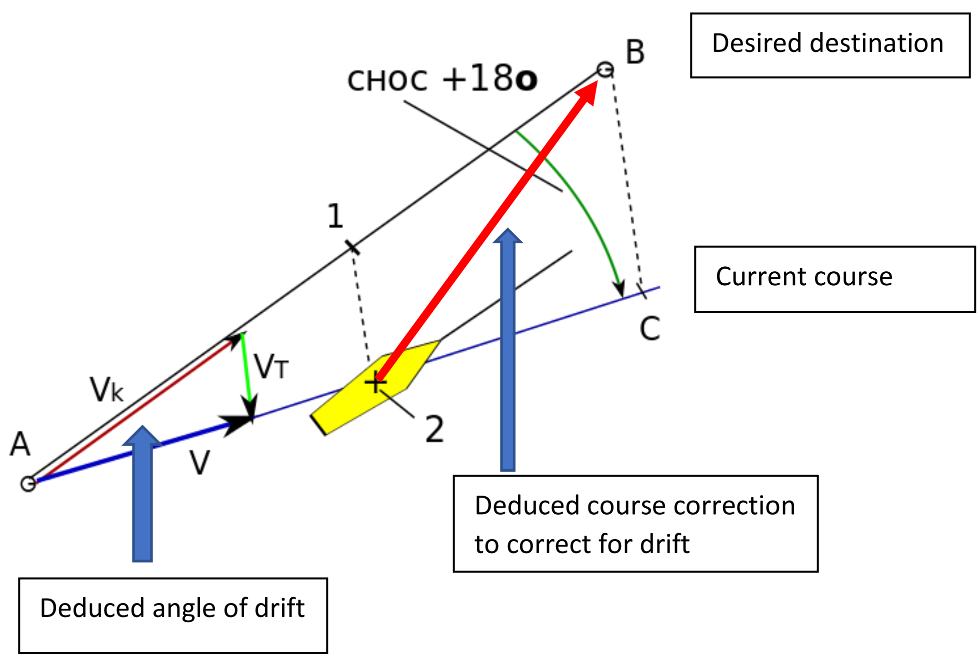
Updated Deduced Predictions of Opponent's Position from Periodic Updates[58]



As we see here, accurately predicting an object's future location using linear physics is difficult to do. First, in the real world—that is, for moving objects such as airplanes or cruise ships—there is the problem of *drift*. An airplane, for example, is likely to drift in predictable ways away from a linear path toward a future destination for many reasons, ranging from the effects of wind (such as side drafts) to pilot inputs. Consequently, to ensure that an object (such as an airplane) gets to its intended destination, accurate dead reckoning involves correcting for drift. This is normally done through triangulation, estimating the likely angle of deviation away from the object's current path, and then inputting corrections (viz. pilot or autopilot inputs) to compensate, drawing the object back toward the intended destination (Figure 17).

Figure 17.

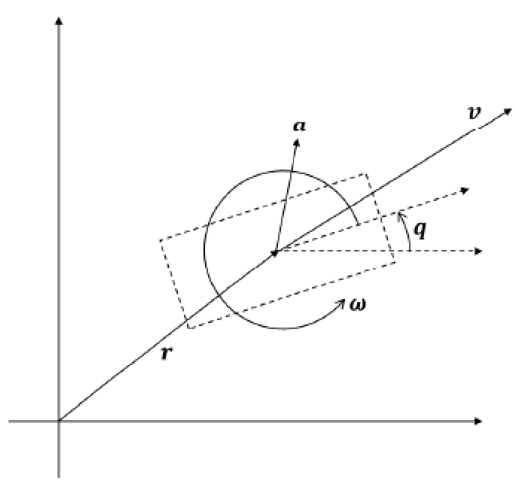
Dead Reckoning Course Correction to Correct for Drift Off Course[59]



Second, moving objects not only drift laterally in space (due to sidewinds); they can also drift from their expected location in space *along a time axis*. For example, by speeding up and slowing down due to headwinds, tailwinds, changes in engine thrust, etc., a plane may end up where it was later expected to be, but 10 minutes behind schedule. Third, wind currents and the like are constantly changing, so to ensure that a plane stays on course, triangulated corrections (for side drift, etc.) must be repeated continuously as conditions change. Proper triangulation to correct for drift, then, requires ongoing, updated measurements of an aircraft's real position, velocity, etc., in relation to the desired destination—which is how aircraft autopilot programs that correct for drift work.[60] Finally, of course, objects in the real world—as well as most online videogame engines—exist not in two spatial dimensions (as in the above examples) but rather three. Thus, in game engines, for example, a player model using dead reckoning is more complicated still (Figure 18).

Figure 18.

Dead Reckoning Player Model for a 3D Game Environment[61]



Because P2P videogames typically involve environments with three spatial dimensions and a fourth temporal dimension, and similar forms of drift occur due to measurement errors caused by lag, latency, and packet loss, programmers utilize a variety of more complex approaches to dead reckoning to achieve 'believable' results.

3. From Dead Reckoning for Believable P2P Games to a Many-Interacting-Simulations Explanation of Quantum Mechanics?

We have already seen how P2P networked simulations reproduce (and hence, plausibly explain) some general features of quantum mechanics—specifically, how (i) all objects in a P2P network exist in a superposition of many different states across the network as a whole, (ii) how every individual computer on the network constantly 'collapses' this superposition at every instant by presenting its user with the object in a determinate state, thus representing a single object simultaneously as (iii) a 'particle' (on every individual computer) but also (iv) as a wave (when summing across the representations of each computer on the network as a whole, such that (v) this 'wave-function' is in essence a probability distribution of where any given user is likely to measure the object to be. These are all fundamental features of quantum mechanics in *our* world, which suggests that P2P networking *may* underlie our world's physics, explaining why our world has these features. What we have not done is demonstrate more precisely that P2P networking reproduces anything like our world's actual quantum wave-function, the substructure of Gaussian wave-packets in our world, the famously bizarre findings of the double-slit experiment, and so on. We will now see that particular approaches to dead reckoning in P2P gaming generate *properties broadly analogous to* a quantum wave-function (§3.1), the specific kind of quantization of information displayed in Gaussian wave packets (§3.2), why particles exhibit wave-interference with themselves as observed in the double-slit experiment (§3.3), why observing individual particles in the double-slit experiment destroys wave-interference

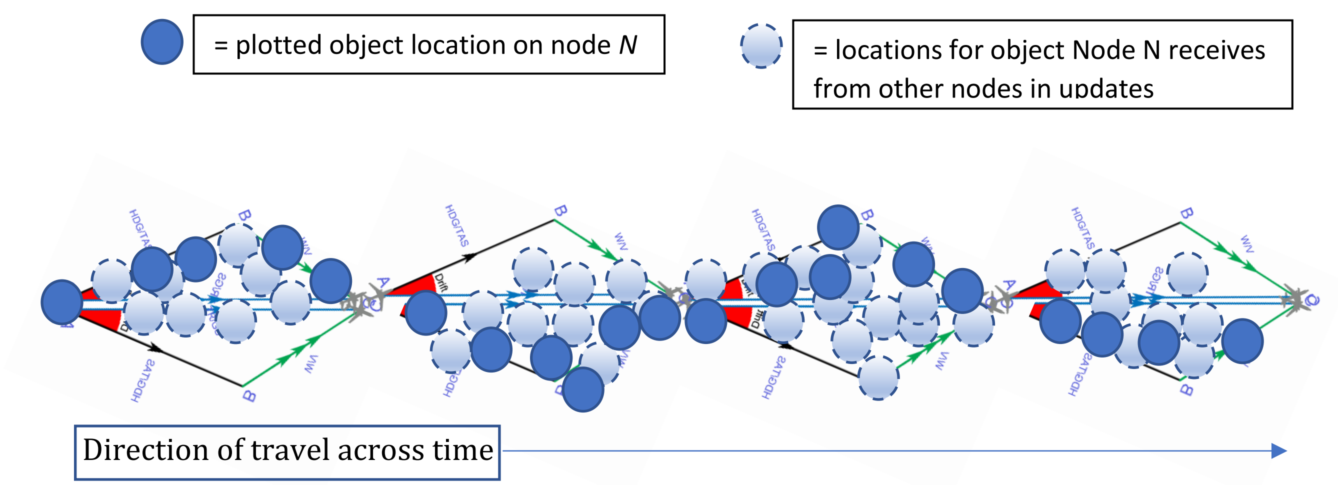
(§3.4), and finally (§3.5) why 'consciousness' appears to play a privileged role in quantum mechanics (qua 'Wigner's friend' thought experiment).

3.1. From Spline-Based Interpolation to a Wave Function

Notice, to begin with, that whereas moving airliners or ships in an ocean tend to drift in particular directions (e.g., downwind), in a P2P simulation the kind of drift we would expect an object to deviate from errors resulting from lag, latency, or loss of informational packets should again be *broadly* random, falling along a normal bell curve (though again, rates of lag can change non-randomly). As such, setting changes in lag aside, dead reckoning corrections in P2P gaming for an object or projectile 'moving in a straight line' should involve the projectile deviating from a center of tendency across nodes (as a result of measurement errors between nodes), and each individual node attempting to *correct* for the deviations from the central tendency that it receives in updates on where 'the object' is from other computers. We can represent this process broadly as depicted in Figure 19.

Figure 19.

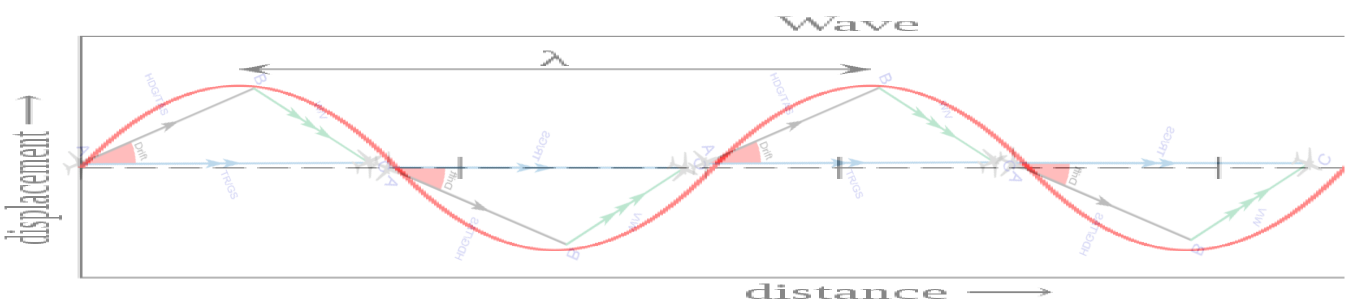
Alternating Dead Reckoning Corrections to Compensate for Random Drift in 2D[62]



Notice, first, that every node on the network will be executing a similar process, adjusting its own plotting of the object's location given the updates it receives from other nodes. Then notice, second, although these are triangular dead reckoning corrections, this sequence has features that are clearly analogous to those of a *wave*. To keep an object 'moving in a straight line' given updates of drift in two dimensions, one needs to assume, first, that the object will deviate vertically from the center line *with a greater distance from the origin* across time (viz. from point A to point B) as measurement errors increase across nodes, and then *correct* for that expected deviation by reducing the object's distance a compensatory amount (viz. the correction from B to the desired position at C), such that each node aims to compensate and reduce measurement errors. These deviations up and down from the center line are akin to *amplitude* changes across time, such that the changes over a given distance approximate a *wavelength* (Figure 20).

Figure 20.

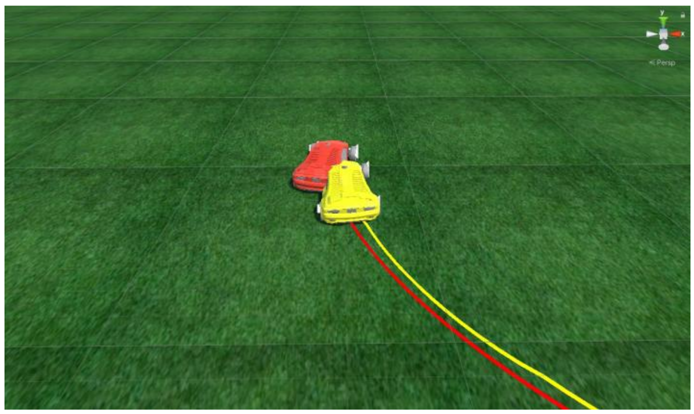
Dead Reckoning Corrections for Random Drift of Moving Object ≈ Propagating Wave[63]



Now, of course, this analogy as described so far is only very rough, as triangulated trajectories are jagged, involving straight lines, whereas waves involve smooth curves. If P2P games used classical dead reckoning by triangulation, players would experience other players and objects in a P2P game moving in a jagged or 'choppy' manner. This would be extremely jarring to players, resulting in a game environment that is nothing like the smooth behavior of objects in space and time that we ordinarily observe in our world. What programmers of P2P games aim for is to achieve this kind of smoothness of motion despite measurement errors/drift caused by lag, latency, and packet loss (Figure 21).

Figure 21.

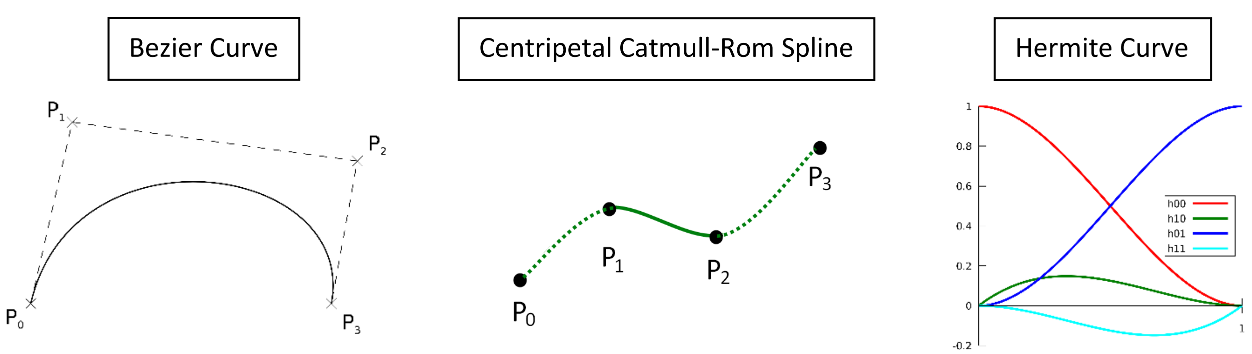
Desirable 'Smooth' Movement in P2P Games[64]



To achieve such 'believable' smooth object paths despite measurement errors, P2P videogame engines utilize several novel programming strategies. First, dead reckoning corrections may use *smooth curves* of various types, including cubic Bézier splines, centripetal Catmull-Rom splines, and Hermite curves[65] (Figure 22).

Figure 22.

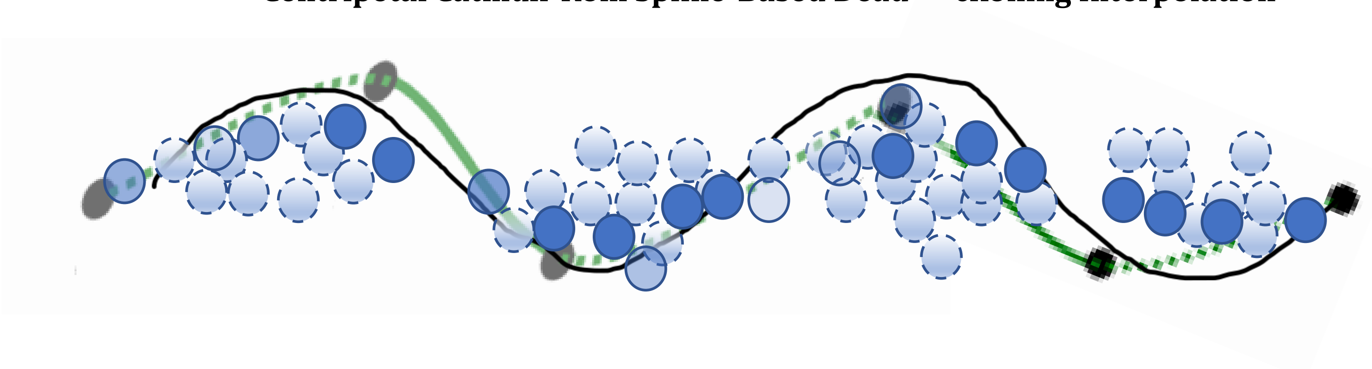
Curves Used in Dead Reckoning P2P Game Engines for Smooth Error Correction[66]



Second, game engineers also use projective linear-velocity and path blending, which are programming strategies where a given computer on the network extrapolates where an object is likely to be by *blending* the different velocities and paths for the object received in previous updates from other computers.[67] We will return to projective velocity and path-blending latter later. For now, notice that if we apply the Centripetal Catmull-Rom spline method to dead reckoning *in place of triangulation*, we get actual wavelengths (Figure 23).

Figure 23.

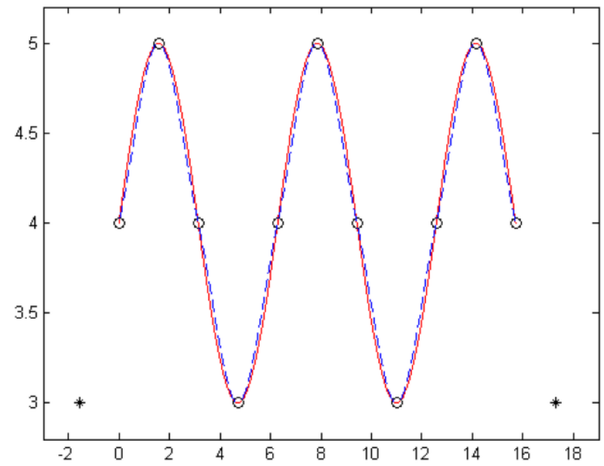
Centripetal Catmull-Rom Spline-Based Dead Reckoning Interpolation



Further, there is an 'optimum cubic α-Catmull-Rom spline interpolation function' in two dimensions, which expresses the function by which 'the smoothest cubic α-Catmull-Rom spline curves can be obtained.'[68] This optimal curve (Figure 24) is, as such, ones that P2P programmers would thus utilize to ensure *the smoothest movement possible* for users in the presence of measurement errors across nodes.

Figure 24.

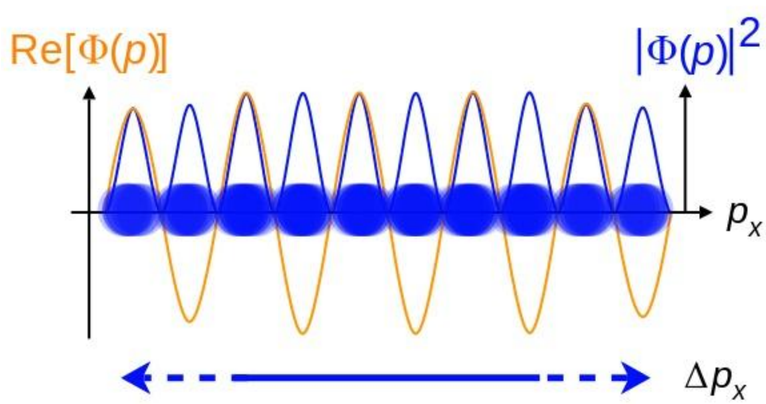
Optimum cubic α-Catmull-Rom spline interpolation function[69]



Notice, next, that there is an obvious sense in which the implementation of this interpolation function should generate wave-like features in momentum space, as in quantum mechanical waves in our universe (Figure 25).

Figure 25.

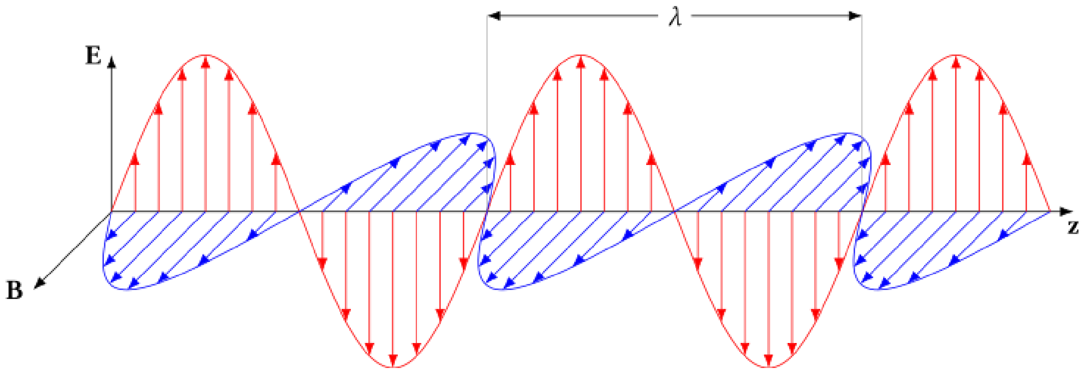
Quantum Waves[70]



Although this is, to be sure, an extremely coarse-grained point of "pattern matching", the conceptual point is nevertheless apt: computationally speaking, using an optimum interpolation function for spline-based dead-reckoning in a P2P gaming environment will produce observations of *something like* quantum waves—particularly when applied four-dimensionally (Figure 26 depicts a classical electromagnetic wave, but the point extends to quantum waves like those depicted in Figure 25).

Figure 26.

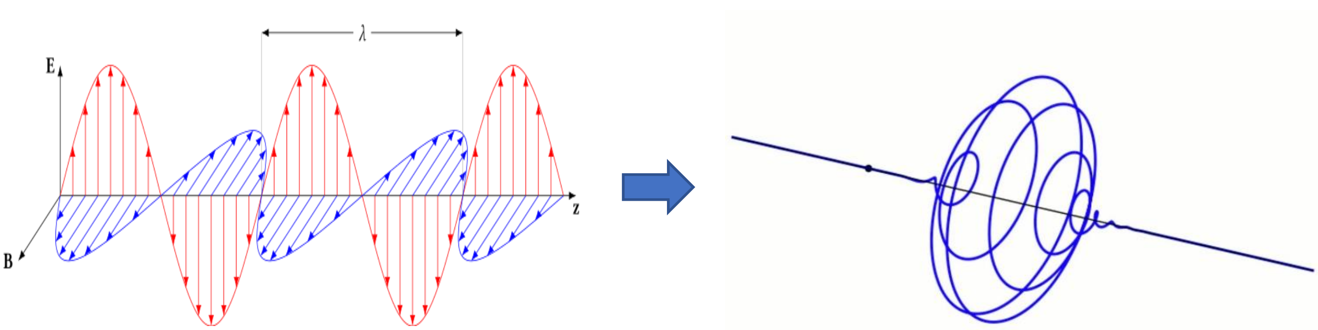
Electromagnetic Wave in Four Dimensions[71]



Further, when a wavelength undulates in four dimensions—vertically across a y-axis and horizontally across an z-axis over time (x-axis)—the result is *rotations* in space-time, which are described by imaginary numbers (Figure 27).

Figure 27.

Rotations of Wave Function in Four Dimensions



While this is not, to be clear, a "derivation of our world's quantum wave function", it does demonstrate a conceptual level that *something like* our world's quantum wave-function should emerge naturally from the computational structure of a P2P simulation involving the above strategies for rendering a 'believable' environment by dead reckoning.

3.2. From Spline-Based Interpolation to Gaussian Wave Packets

The next thing to note is that all information in computer networks is sent between computers in small, distinct packets[72], much like energy in our world is always emitted in distinct packets known as quanta.[73] To begin with a simple example, Internet Control Message Protocol (ICMP) packets are *error-reporting* strings of data that network devices (such as routers) send to computers on a network to generate error messages.

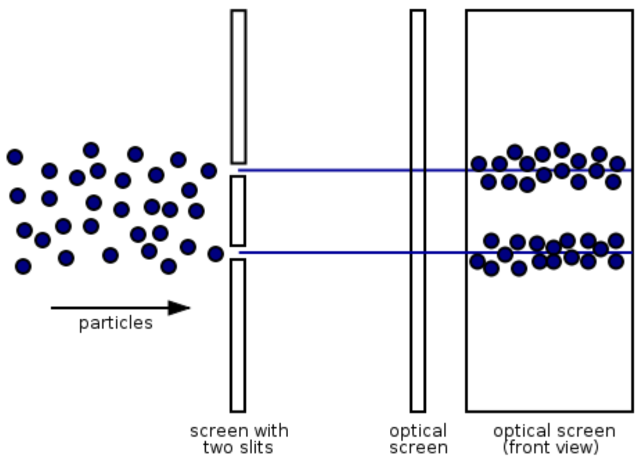
Notice, next, that ICMP packets have both a *superstructure* and *substructure*. On the one hand, the *entire packet* is a distinct body of code (33 bits). Yet, the packet itself has a *substructure*: header information, a 'pointer' identifying the error, 'supplementary information', and so on. The kind of information that P2P videogames send each other is far more complex, but the principle is the same. Networked computers function by sending bursts of self-contained data packets to other computers on the networks—yet, every data packet contains various substructures (such as one substructure indicating where a projectile is, another substructure indicating its velocity, etc.). Consequently, due to lag, latency, and packet loss, these superstructures and substructures should *both* generate wave-like dynamics approximating normal distributions. That is, summing across P2P nodes, there should be normal variance (viz., a bell curve) for each *microstructure* encoded in a data packet (e.g., projectile-location, velocity, etc.), but also for the entire *macrostructure* (e.g., the entire 'data envelope')—just as we see in Gaussian wave packets in our world.

3.3. From Velocity and Path Blending to the Double-Slit Experiment

Of the many strange features of quantum mechanics that have been experimentally established, the findings of the double-slit experiment[74] are perhaps the most bizarre. In this experiment, particles of light are passed through two slits in a barrier one at a time. If light simply consisted of particles, we would expect to observe the pattern in Figure 28.

Figure 28.

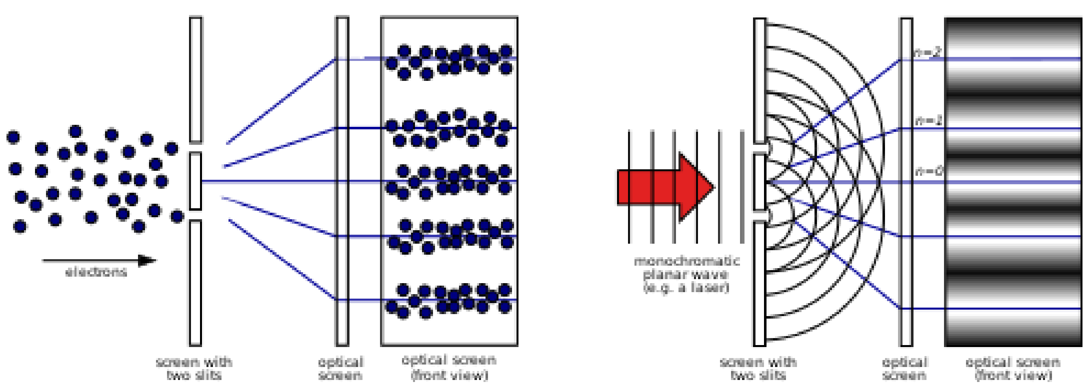
Expected Behavior of Particles Passing Through Two Slits[75]



However, when light is passed through *two* slits, wave inference is observed instead (Figure 29).

Figure 29.

Wave Interference Patterns in the Double-Slit Experiment[76]



Importantly, this occurs regardless of whether one fires just one particle through the slits at given time or many particles all at once. It is almost as though the particle *splits in two*, travels through both slits, the particles interfere with each other, and then rejoin before hitting the wall. Further, this behavior—'single' objects appearing to somehow interfere with themselves qua wave—has been found for photons, electrons, atoms, and molecules.[77]

If this were not strange enough, when a detector is set up to measure *which* slit the particle moves through, the wave-inference pattern immediately disappears, the particle is always seen as moving through only *one* slit, and the pattern on the wall behind the slits reverts to how particles behave when shot through a *single* slit. Finally, if the detector is turned off again, the wave-interference immediately reemerges. All of this suggests that photons and other objects in our world exist *as* waves spread out across space and time (and across multiple slits) when they are *not* being directly detected, but as *particles* the moment they are measured. Although Einstein (a critic of quantum mechanics) once scoffed about these implications, 'Do you really think the moon isn't there if you aren't looking at it?'[78], quantum mechanics really does appear to establish that everything somehow exists in a wavelike state *until* they are measured, at which point they 'collapse' to determinate states (in this case, passing through one slit *or* the other, but not both).

Can the double-slit experiment be explained by dead reckoning approaches to 'believable' P2P game engines? Indeed, it may. First, consider how game engineers program games to *only render observed content* as a means of minimizing computational complexity:

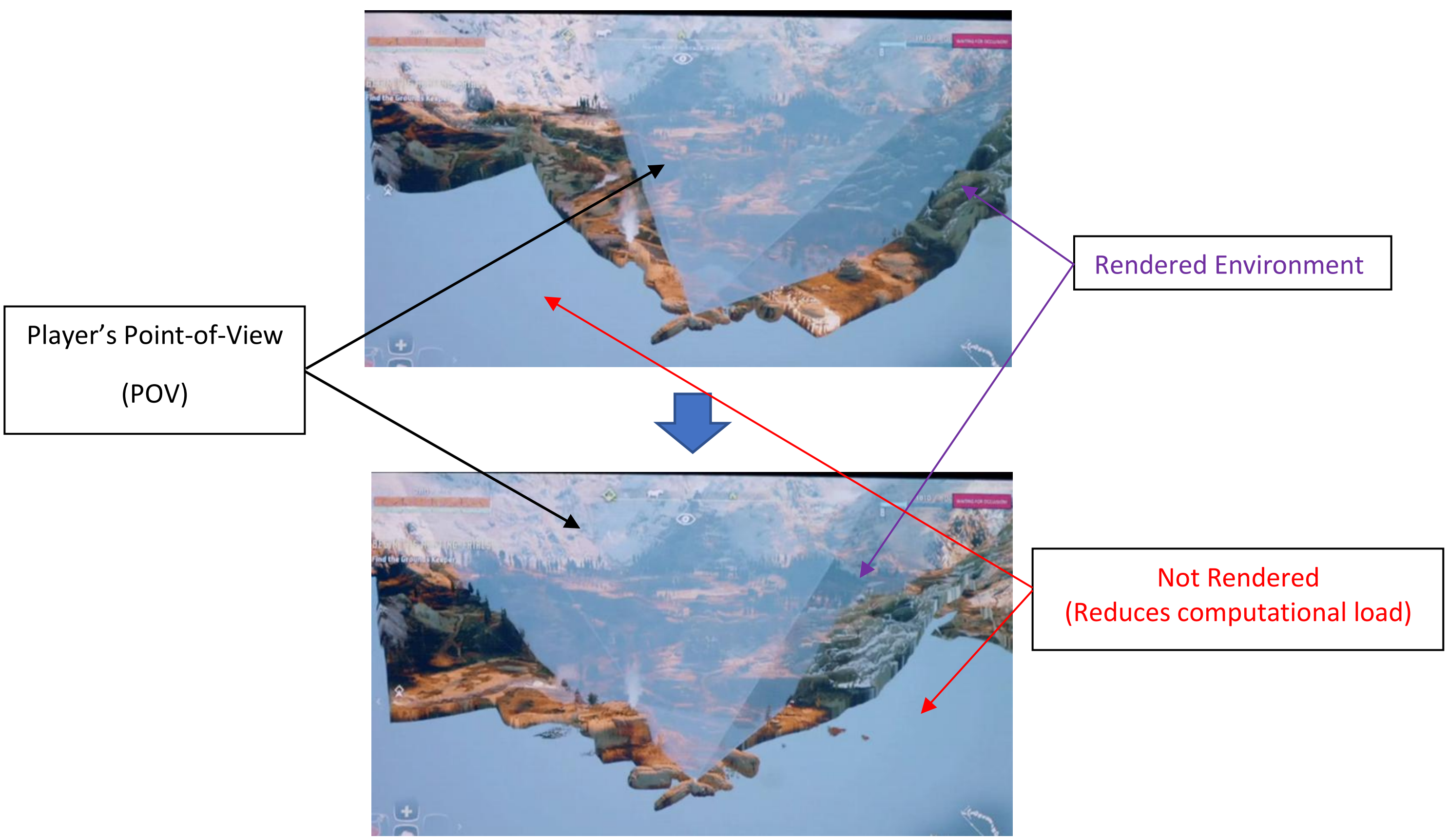
> It is … now well understood, in the domain of game development, that low computational complexity requires rendering/displaying content only when observed by a player. Recent games, such as *No-Man's Sky* and *Boundless*, have shown

that vast open universes (potentially including over 18 quintillion planets with their own sets of flora and fauna …) are made feasable [sic] by creating content, only at the moment the corresponding information becomes available for observation by a player, through *randomized generation techniques* (such as procedural generation).[79]

In other words, rather than constantly rendering exactly where all 18 quintillion planets in a game environment are, games *only* render objects for a user, giving them a determinate location and state on a given computer *when observed by that player* (Figure 30).

Figure 30.

Game Only Rendering Observed Content in Player's Point-of-View (POV)[80]



Second, the same computational issues arise with respect to the representation of macro- and micro- states of a game environment. For example, suppose that to make a 'believable' world, game developers wanted to ensure that every macro-object in the game—such as projectiles, persons, and planets—are composed in a well-organized manner by smaller elements, call them 'particles.' In much the same way that it is computationally intractable for a game engine to constantly represent every detail of 18 quadrillion planets, so too is it intractable for a game engine to constantly represent every detail of the 7 octillion atoms or so in a single human body[81], let alone the 3.28×10^{80} particles in the observable Universe.[82] The obvious way to handle this problem in game engineering is to 'to render reality only at the moment the corresponding information becomes available for observation by a conscious observer (a player)', such that the, '*resolution/granularity* of the rendering would be adjusted to the level of perception of the observer.'[83] In other words, the only feasible way for a game engine (including P2P networked game engines) to represent microparticles in a game environment is *not to render them* until directly observed.

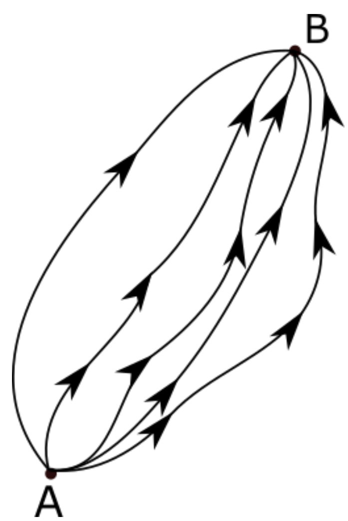
The next question is how a *P2P* game engineer could make such an environment broadly 'believable' at a macro-level *and* micro-level, given the manner in which different computers on a P2P network do not "agree" with each other on precisely where (A) macro-objects (such a projectile) *are* located at any given instant or (B) where the micro-objects that constitute those macro-objects (such as particles) *will* be when they are actually rendered. Notice that a big part of the programming challenge here is to *relate* the macro- and micro-parts of the game world in a consistent way. If, for example, a player used a simulated microscope to 'zoom in' on a projectile to 'look at its constitutive particles', their game will at that instant begin to render those particles as being *at* determinate places 'inside; the projectile. However, as we saw earlier, different computers on a P2P network will represent the macro-object (the projectile itself) as *being in slightly different places* (with slightly different velocities, etc.) than other computers. Consequently, when a different player 'zooms in' to view the constitutive particles of one and the same projectile, their computer will then render those microparticles in determinate places from their point of view—but again, not the *same* states as other players.

Our next point is critical: differences in the observed location of a macro-object (such as a projectile) that are 'miniscule' from a macro-perspective (such as a 1-millimeter divergence) imply *massive* divergences for micro-objects, such as the locations of those macro-objects' constitutive particles. To see how, suppose that my computer on a P2P network represents a projectile in one location, and your computer represents it 1-millimeter away, in a slightly different location. From a macro-perspective, this difference will be barely noticeable to either of us. Now consider, however, what our respective computers must represent a single particle in the projectile *if observed/rendered.* An electron has a diameter of somewhere around 0.000000000001mm.[84] Suppose next that engineers program projectiles to be partially constituted by 'electrons' with this size, such that when a player 'zooms' with a simulated microscope, their computer will *render* the 'electron' in a given position with that size. If two different computers on a P2P network represent this macro-object (the projectile) in two different places just 1mm apart, then when deciding where to render a single 'electron' in that projectile (if zoomed in upon), to ensure that the 'electron' is measured to be *within* the 1mm radius of where the projectile is located *across* the two nodes, those two computers will need to represent it as having approximately 10^{13} different likely locations *spanning an entire millimeter of space*.

In other words, to ensure that 'micro-observations' align with observations of macro-objects across nodes, different computers will have to not assign *any* determinate locations to micro-objects until rendered (as this would again be computationally intractable), but instead represent them as having a *vast number of possible paths through space and time*, as in a path integral description of quantum mechanics (Figure 31).[85]

Figure 31.

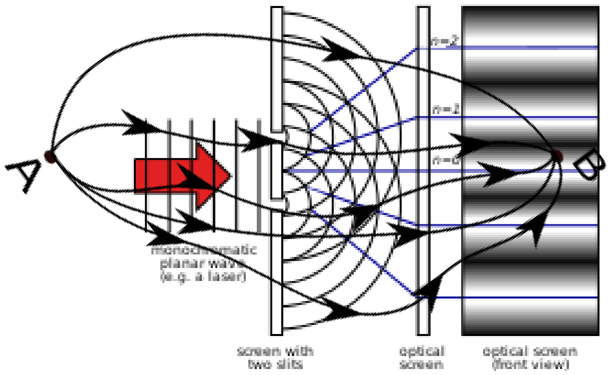
Path Integral Description of Quantum Mechanics[86]



We are now in a position to potentially explain the double-slit experiment results in terms of P2P networking and dead reckoning. First, if different computers on a P2P network represent particles as *likely* being in many different possible places in space *before being rendered/observed*, then when a macro-object (such as a proton gun) emits light toward a barrier with two slits, each computer on the network will have to represent the possible locations of a *single* particle something along the lines of Figure 32.

Figure 32.

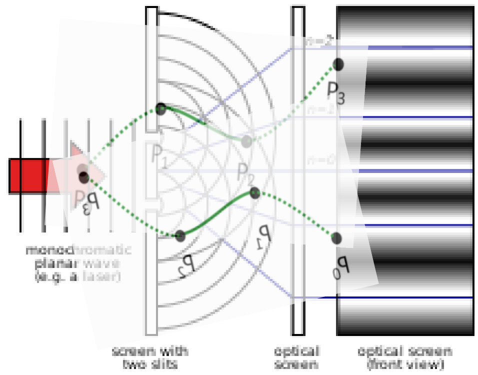
P2P Nodes Representing Where a Given "Particle" is Likely to Be Before Rendering[87]



In other words, *before* rendering where a particle is (when merely taking macro-level measurements), each computer on a P2P network will have to represent a single micro-particle as likely taking *many different paths* through space and time from a given source to destination. However, as we have seen, for this to be done in a computationally tractable way, no computer on the network—nor the network as a whole—can actually represent the particles as *actually taking* all of those paths (as representing each path as actual would require an immense amount of computing resources). What programmers would have to do is to take a *shortcut* that generates the same probability distribution for where to render a micro-object (viz. many possible paths) without rendering those paths. The answer of how to do so is for each machine on the network to apply curved dead reckoning solutions (the same ones we examined earlier) to where particles are likely to be, but *without rendering where the particle is* (Figure 33).

Figure 33.

Catmull-Rom Dead Reckoning Shortcut to Reduce Complexity[88]



In other words, the only computationally feasible way to realize a 'believable' P2P game environment that suitably aligns the properties of *rendered* macro-objects (such as projectiles or large shadows of light) with where micro-objects (such as individual particles) *should* be rendered (if 'zoomed in' upon) is to:

1. ***Render the likely positions of micro-objects as dead-reckoning wavelengths*** when it is computationally intractable to render an exceedingly large number of individual object paths to account for macro-object behavior.
2. ***But, the moment any player 'zooms in'*** to measure such a particular micro-object, render it as being in a particular location with determinate properties (qua 'point particle').

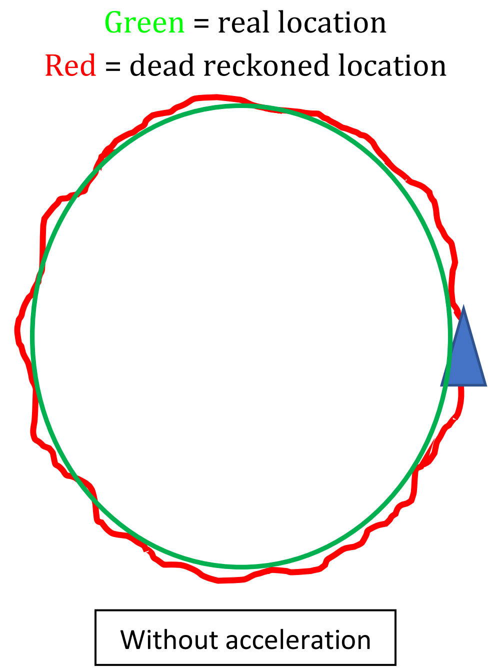
But, this is more or less what the double-slit experiment indicates: that particles, *at least when traversing two slits*, are 'rendered' in our reality as *waves* that (in effect) represent a single particle as (likely) 'traveling through both slits'—but, the moment any detector seeks to measure exactly where the particle is located, it is always 'rendered' as being *at* a particular point (traveling through one slit or the other, but not both).

Two further issues also need explaining. First, we need to explain why particles are *not* rendered as waves when traveling through just one slit. Second, we need to explain the double-slit wave-interference pattern that occurs when individual particles are not measured by a detector. The first issue here is relatively straightforward to address. Videogame programmers can program game engines to utilize computational shortcuts when, but only when, not using the shortcut would overtax the system. In the case of the double-slit experiment, the explanation here is whatever network of powerful computers comprises our reality, computers on that network *are* able to render individual particles passing through a single slit without utilizing the above shortcut, and it is only when computers attempt to keep track of far more complex particle paths (viz. two slits) that the shortcut is triggered. One interesting feature of this explanation is that we may be able to extrapolate the computational limits of the P2P networked computers (and computational resources and limitations of the network as a whole) from this feature of quantum mechanics—that is, by explaining what computational limitations such a system would likely have for programmers to choose this particular shortcut under the specific types of conditions of the two-slit experiment (but not the one-slit variant).

Thus far, we have examined how dead reckoning strategies to ensure a 'believable' gaming environment use Cubic Bézier splines and Hermite curves to ensure smooth-looking paths for objects, and how these programming strategies generate wave-like dynamics. However, these approaches notoriously generate wave-like undulations for macro-objects (such as simulated cars). Left without further correction, these types of oscillations result in unrealistic oscillations of objects in P2P environments (Figure 34).

Figure 34.

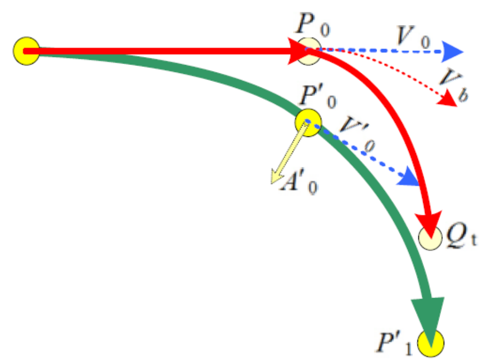
Oscillations of Object Paths from Cubic Bézier Spline Interpolation[89]



If you or I drove a car in a circle (real position) but other observers (due to dead reckoning) viewed one as driving in one of the undulating red shapes above, then our 'shared reality' would not be believable. We would observe macro-objects appearing to do very different things. This problem is dealt with in P2P gaming through two additional approaches to dead reckoning, known as *projective velocity blending* and *path blending*. Projective velocity blending involves any given computer taking a measurement of a given object (e.g., my game console) blending—or adopting the average value—between its own representation of the object's velocity and the velocity reports it receives about the object from other nodes on the network. Figure 35 displays the simple case of *linear* projective velocity blending[90] between where an object is represented by one computer (P_0) with object velocity reported by a second computer (P'_0)—such that the next velocity plotted (Q_t) comprises the average value in between.

Figure 35.

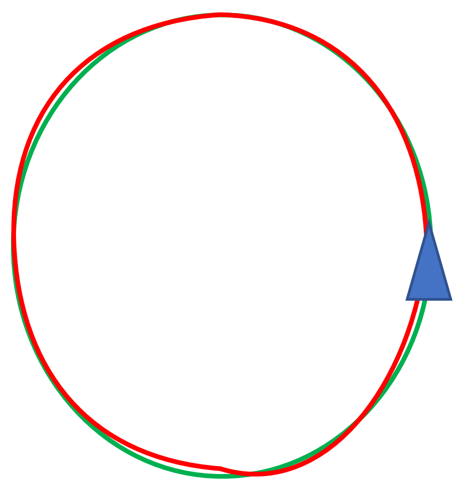
Dead Reckoning with Linear Projective Velocity Blending[91]



Projective velocity blending results in more 'believable' object paths (Figure 36)—in the above case, a vehicle moving *approximately* in a smooth circle without large undulations.

Figure 36.

Object Path with Projective Velocity Blending[92]
Green = real location
Red = dead reckoned location



This, however, is merely for objects with simple trajectories (such as a circle). For complex object trajectories, a more complex approach—*path blending via neural networks*[93]—is utilized. As Figure 37 illustrates, neural network path blending is just like velocity blending, except that it incorporates an additional variable—'displacement between the current and prediction *positions*'[94]—along with velocity.

Figure 37.

Overview of Path Blending Neural Network[95]

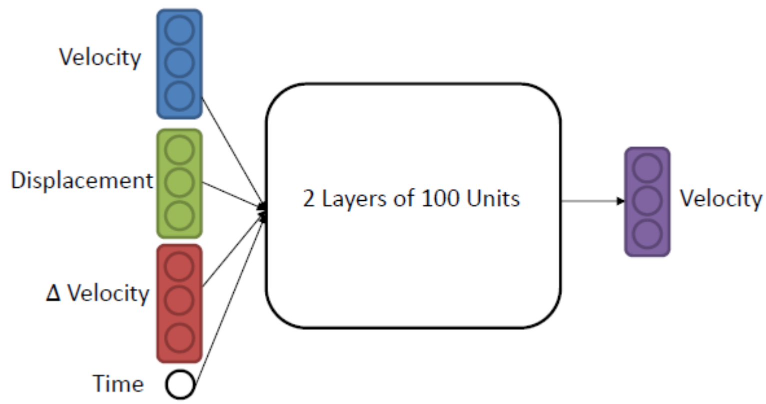
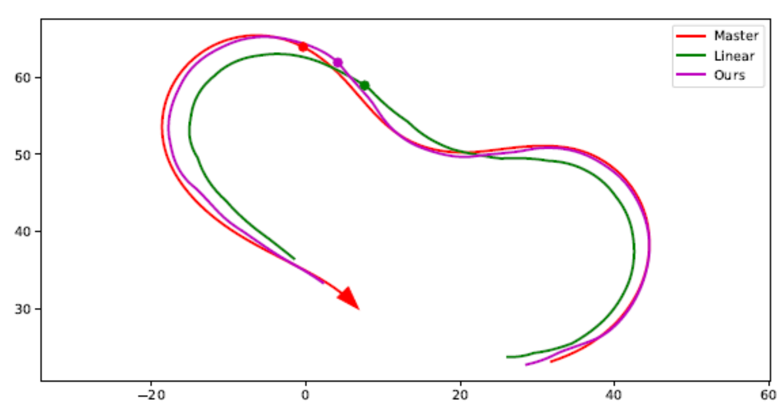


Figure 38 illustrates how accurately a state-of-the-art neural network path blending approach ('Ours') predicts object locations ('Master') compared to linear velocity blending ('Linear'), even with untuned parameters—that is, without engineers tuning the algorithm to produce desired paths—as well as a relatively long 300ms message interval between computers (an interval far longer than P2P games typically communicate, i.e., <100ms).

Figure 38.

Accuracy of Path Blending Neural Network Dead Reckoning[96]

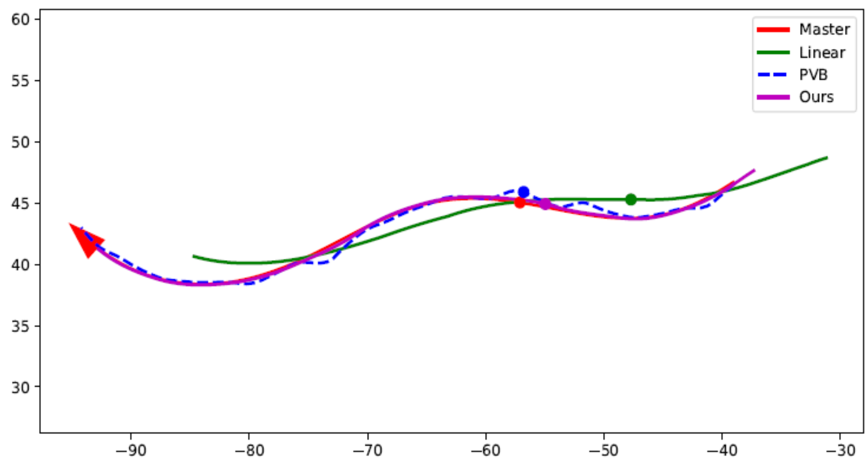


"Results of path tracking blending with untuned parameters and a message interval of 300ms"

Predictive path blending is highly accurate, resulting in predicted object locations that match closely with the actual object location recorded in updates by other computers on the network. Path blending has even greater accuracy when shorter message intervals are used between nodes (e.g., 200ms), and when its parameters are tuned by the programmer to improve accuracy (Figure 39).

Figure 39.

Accuracy of Path Blending with Tuned Parameters and 200ms Message Interval[97]



As we see here, the path blended approach with these characteristics ('Ours') matches the *actual* path reported by other nodes ('Master') almost perfectly. This demonstration is important because it demonstrates that to provide a 'believable' game environment—one in which all players observe approximately the same object behavior, with smooth object movement (rather than choppy or undulating deviations from what one would otherwise expect), path blending must be utilized: each computer on a P2P network must *blend* object paths and velocities that it receives in updates from other computers to plot where the object should be rendered at the next instant.

When combined with our earlier explanation of the double-slit experiment, this explains the constructive and destructive wave interference observed in the two-slit experiment when there is no particle detector present. Recall how, to reduce computational complexity, game designers would need to program P2P games to render spline-based waves rather than every actual particle path. If this is what each individual computer renders (when individual particles are not being measured)—again, as a shortcut to avoid having to render every particle path (which is computationally infeasible)—then, following the same measurement errors that arise in P2P networking more generally, different computers should render *these very waves* in slightly different locations. But, as we saw earlier, the 'overlap' between what different computers on a P2P network render should fall along a normal bell curve, qua Gaussian wave. That is, in regions closer to the central tendency of where a object (such as a projectile) is represented, there should be substantially less error between where one computer represents the object and whether other computers on the network do. This means that if we program the shortcut we have been examining, two or more computers should render waves that overlap well along some trajectories but *fail* to overlap (or get out of phase across machines) further away from those areas of central tendency.

In other words, once we adopt the wave-rendering as a computational shortcut in P2P environment (to avoid the infeasibility of rendering every microparticle) and we combine this with the natural normal distribution of measurement errors that occur in P2P networking, it follows that across any area where this function is being rendered—that is, conditions where the computers need to render certain macro-effects (such as shadows of light) but cannot feasibly do this by rendering every particle (as in double-slit cases), the waves that different computers render will be *completely in phase* close to areas of central tendencies and *completely out of phase* further away from the central tendency.

The point is this: if we combine this with neural network path-blending—that is, a dead reckoning approach where *each computer* blends (i) the path they are rendering with (ii) the paths that other computers on the network are sending in—then in areas of the simulation where there is good overlap (close to areas of central tendency), the path blended solution will be *constructive* wave-interference. However, in areas away from the central tendency, where waves are out of phase across machines on the network (again, due to these regions being outliers where measurement errors are greater), the path blended solution—taking the average between the waves out of phase—will be *destructive* wave interference. But this is exactly what is observed in the double-slit experiment.

So, we have explained how an entirely natural approach to handling problems of measurement error correction and computational complexity in 'believable' P2P game design—(i) rendering only *observed* objects, (ii) avoiding rendering *micro-objects* that would outstrip the computational resources of any feasible computational system *until* they need to be rendered when directly observed, (iii) rendering *dead-reckoning wave corrections* as computationally feasible simplifications to keep track of where micro-objects are likely to be given the macro-world that is rendered at a given time, and (iv) combining spline-based interpolation with *predictive velocity and path blending*—together explain why we observe the bizarre results we do in the double-slit experiment in our world. The results of the experiment are just what one would expect if *we* are living in a vast P2P network computer simulation utilizing state-of-the-art solutions to resolving measurement errors in a way that is as 'believable' as possible given computational constraints.

3.4. From Dead Reckoning for Believable P2P Gaming to Wigner's Friend

Eugene Wigner gave a famous thought experiment, now known as Wigner's Friend, that he took to show that consciousness plays a privileged role in quantum mechanics: specifically, that only *it* can collapse the wave-function.[98] Wigner's example is a variation of another famous problem, known as Schrödinger's Cat.[99] In Schrödinger's example, we are to imagine a cat inside a sealed steel chamber with a Geiger counter, a tiny bit of radioactive substance, and a small flask of hydrocyanic acid. During the hour in which the cat is in the chamber, the substance has a 50% probability of one its atoms decaying. If the atom does decay, it will trigger the Geiger counter to break the flask of acid, which will kill the cat. But, if this does not occur, the cat will live. Given that the radioactive system is a quantum system, quantum mechanics holds that these two scenarios—the one in which the cat dies and the other in which the cat does not die—exist in a *superposition*. The question then is when, and how, the superposition stops existing and one of the scenarios will occur ('collapsing' the wave function). According to Copenhagen interpretation of quantum mechanics (which a recent informal survey indicates to be the most popular interpretation among specialists by a significant margin[100]), Schrödinger's Cat is both alive *and* dead until the box is opened and the quantum system inside the box is observed.[101] Schrödinger himself took this seemingly paradoxical result to challenge the Copenhagen interpretation, and alternative interpretations of quantum mechanics have been advanced as ways of avoiding this and other apparent paradoxes.

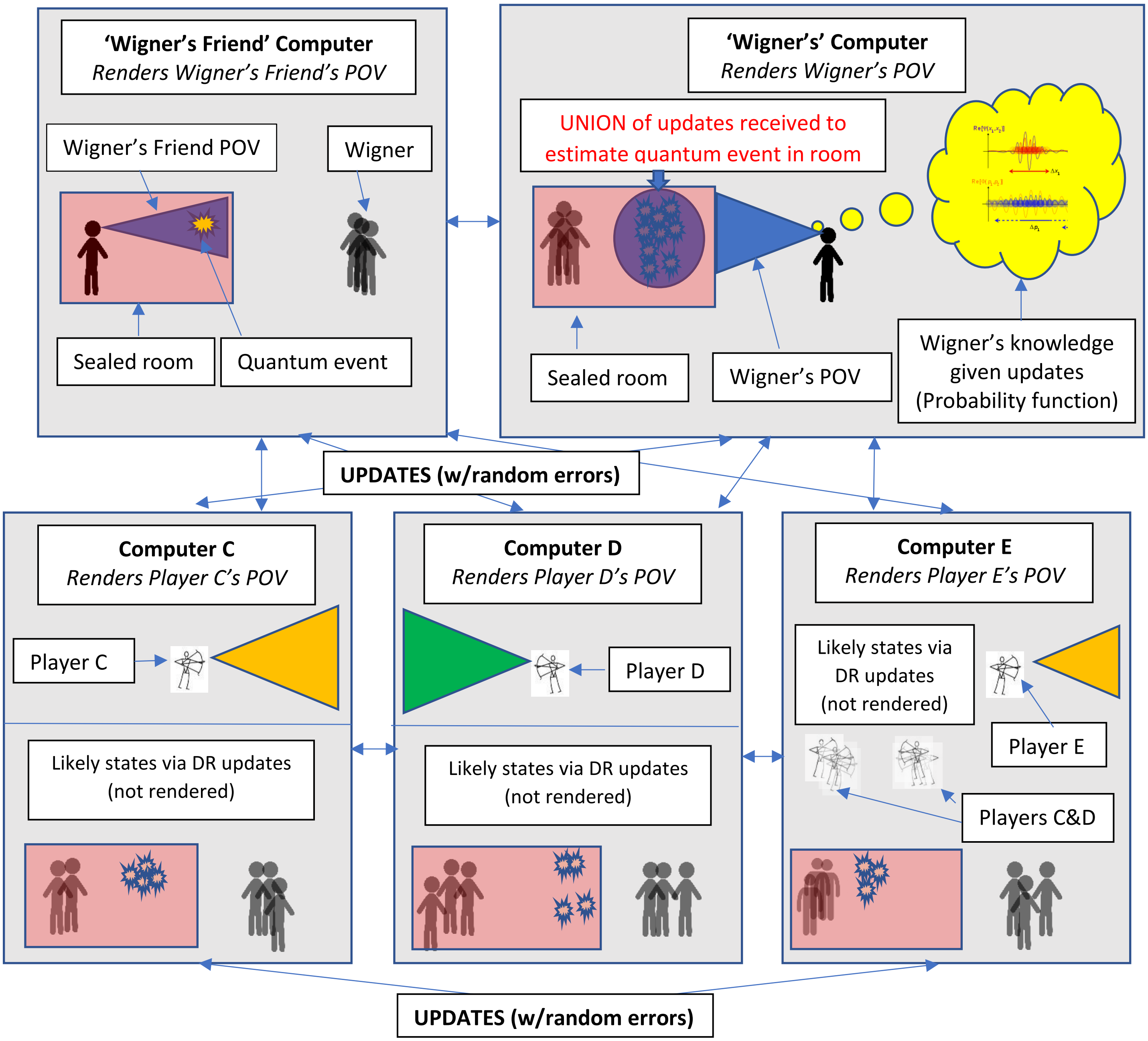
Wigner's example is a variation on this scenario—one in which, instead of a cat with poison inside the box, there is a human being (an 'observer') within a sealed room who will either observe a flash on a screen if a photon hits the screen or observe no flash if the photon does not. Wigner points out that, according to quantum mechanics, this person inside the room—Wigner's friend—will never observe a superposition between these two states: they will either observe a flash on the screen or not, but *not both*. Next, however, we are supposed to imagine Wigner himself outside of the room—a second observer who, just like in the original Schrödinger's Cat case, cannot observe what has happened in the room until it is opened. According to quantum mechanics, from Wigner's standpoint as an outside observer, what happens in the room *is* in a superposition until Wigner can see inside, at which point—upon measuring the system—Wigner will find one of the two results to have happened but not both. For Wigner, this example demonstrates that consciousness itself (and only consciousness) collapses the superposition described by the wave-function. For, as we see here, conscious observers never observe superpositions. Wigner's friend's consciousness collapses the wave-function within the box *for the friend*, but it is only when Wigner peers inside the room that the wave-function is collapsed *for Wigner*. This suggests that the world itself exists in contradictory states during the intervening period for Wigner and his friend. From Wigner's standpoint outside of the room, what occurs in the room *in fact* exists in a superposition: a measurement cannot have taken place inside the room until Wigner peers inside. But, from his friend's standpoint inside the room, a measurement *has* been taken—and a clear outcome observed (a flash or not a flash)—before Wigner takes a measurement. Finally, and perhaps most astonishingly of all, these implications appear to have been experimentally verified: relative to a couple of background assumptions (concerning 'locality and free choice'), an experimental examination of violations of Bell's inequality reveals that two observers can indeed experience 'different realities.'[102]

We can explain these results by P2P networking and dead reckoning in several steps. We may recall, first, that in P2P gaming, every player *does* experience their own distinct 'reality', such that at any given point in time they never *will* experience a superposition, though from the standpoint of the network as a whole, all of their realities will jointly *constitute* a superposition. Secondly, however, as we have seen P2P game programmers utilize partial rendering strategies—where only part of an environment is actually rendered (namely, the environment being observed by a given player)—to save processing power, a strategy necessitated by limitations in processing power. Next, we saw that the natural way to do this while ensuring a believable environment for all players is for the game engine to simply apply dead reckoning probability functions to non-rendered phenomena, so that when any player does render the relevant object(s), the object will be rendered as having a location and properties *that approximate* where any other player will measure the object if they become in a position to render it (by bringing the object within their point-of-view/POV, including by 'zooming in').

Here, then, is what we have. Let us treat one computer as 'Wigner's friend's computer', where this computer renders the *point-of-view* (POV) of Wigner's friend as the game-player (e.g., a flash of flight or lack thereof caused by a particle hitting or not hitting a detector, respectively). Then let us treat a second computer as rendering *Wigner's* POV. Finally, since these two computers communicate with other computers in a P2P network, let us image three other computers rendering three other players' POVs some distance removed from Wigner and his friend's environment. In that case, given how we have seen partial rendering with dead reckoning for believable P2P games to work, Wigner's friend *will* observe a determinate event inside their room (e.g., a flash of light). However, from the standpoint of every player outside of the room, including Wigner—and from the standpoint of the *union* of all of the computers that Wigner's is communicating with—what occurs in Wigner's friend's room will be a superposition described by a wave function (Figure 40).

Figure 40.

Dead Reckoning (DR) Explanation of Wigner's Friend[103]



In short, when dead reckoning is used to calculate likely object positions given updates from all other computers on the network, Wigner will receive updates that the object is likely in a number of places simultaneously—but, because the actual quantum event (the flash in the sealed room) is only in Wigner's friend's POV, *only Wigner's friend's computer* will render the event with determinate properties (as a single flash). Dead reckoning approaches to P2P gaming thus explain the privileged role that 'consciousness' has in 'collapsing the wave function.' P2P games only render objects as having determinate properties *within the POV of the individual gamer*. This means that all other gamers on the network who cannot view those events will never render them with determinate qualities (e.g., eigenstates) until they look inside, at which point *their* computers will render what happened inside the room, 'collapsing' the wave-function that describes these dynamics for *them* at a later point (and in a different way) than Wigner's friend's computer 'collapsed' it for him while he was inside of the room. This also explains why no one—neither Wigner nor Wigner's friend—ever *actually* experiences objects in superpositions. Wigner's friend always experiences determinate events within the room (e.g., a flash at one point rather than another) because his computer is *rendering* those events. All other computers on the network will represent those events as in a probabilistic superposition until they view the system themselves, at which point their computer will render it *for them*. This explains how it *is* our point-of-view—that is, our 'conscious first-personal perspective'—that constantly 'collapses' object superpositions so that we never *experience* them.

4. How Extant Interpretations of Quantum Mechanics May Each Be Partly True

Notice that, if correct, this paper's model may explain quantum phenomena in a way that no other interpretation does. Rather than taking quantum phenomena as given—as fundamental physical features of our world—our new interpretation gives a deeper computational explanation of why our world *has* those features in the first place. Here, we believe, is a helpful analogy. Imagine that we lived on the surface of a giant clock, and we witnessed an hour hand, minute hand, and second hand traversing the sky. After measuring their behavior, we might put together a theory of 'fundamental physics' for this world comprised by equations describing the surface features and behavior of the hour hand, further equations describing the minute hand, a third set of equations describing the second hand—and finally, a *unifying* equation explaining how their behavior relates (viz., an equation describing how, every time the second hand traverses 360 degrees, the minute hand advances 1/60th of that same circumference, and how every time the minute hand traverses 360 degrees, the hour hand advances 1/60th of the circumference). Assuming we had such a 'fundamental physical theory' in hand, we might aim to interpret its ontological significance. Are there deeper 'hidden variables' not observable in the physical world that somehow *relate* the behavior of the hour, minute, and second hands? Or, is it just a brute law of nature that the different hands behave in the strangely well-organized manner that they do? Here, we might speculate endlessly. But, surely some particular answers are better than others. Given that the physics of the world question behaves just like that a clock, there are some grounds for that the world has a deeper functional structure: namely, the structure of a clock—structures that explain why the 'hands in the sky' move as they do. That is, if we observed the world behaving *just like a clock*, then we would have abductive grounds for believing that the world *is* a clock—that we live in a 'Clock World.' After all, this interpretation of the behavior of 'Clock World's fundamental physics' provides a deeper, unified functional explanation of the world's behavior that other interpretations do not—

one that, as an inference to the best explanation, suggests that the world's fundamental physics is not fundamental after all, but rather emergent from a deeper functional structure. Our new interpretation of quantum mechanics is just like this. Further, in addition to providing a unified explanation of the existence of quantum phenomena and their many otherwise bizarre features, our new interpretation reveals that many existing interpretations of quantum phenomena contain some elements of truth.

Consider the Copenhagen interpretation. There is, surprisingly, no definitive statement of exactly what the Copenhagen interpretation is, as there are fundamental disagreements between its proponents (particularly Bohr and Heisenberg).[104] However, as a very rough first approximation, this interpretation holds that quantum particles exist in all of their possible states (given by the quantum wave-function) at once in a superposition until a measurement is taken, at which point the wave-function *indeterministically* 'collapses' such that the particle is observed to *approximate* classical behavior, viz., having a *near* determinate state (though, following Heisenberg's uncertainty principle[105], it remains impossible to know both the position and speed of a particle with perfect accuracy). Our new interpretation/explanation of quantum mechanics shows there to be a real element of truth to the Copenhagen interpretation. For, as we have seen, in believable approaches to dead reckoning in P2P gaming, all objects *do* exist in a constant superposition from the standpoint of the network as a whole, and at every instant that a given computer on the network takes a measurement—rendering the local environment for its user within their point-of-view—*that* computer will measure the object(s) in that environment as having near-determinate properties (e.g. a definitive location, though due to measurement errors and dead reckoning, such an observer will *not* be able to predict the object's next location—and so, direction and velocity—with absolute certainty, just as is the case for quantum phenomena in our world). So, if our explanation is correct, the Copenhagen interpretation is partly true: it is just not the whole truth. Notice, finally, that our interpretation provides a new resolution to perhaps the most fundamental problem with the Copenhagen interpretation: the *measurement problem*.[106] In brief, the Copenhagen interpretation requires invoking observers or measuring devices in a *classical* (non-quantum) domain to 'collapse' the wave-function—yet, according to quantum mechanics, observers/measuring devices are *themselves* quantum devices, not classical ones. Our interpretation provides a novel solution: every computer on a P2P network *renders* a classical domain (i.e. objects having near-determinate properties) at every instant, which it does through *dead reckoning* on the basis of updates from other computers on the network—yet, that very act of measurement engages with the network *as a whole* (by receiving messages from many different nodes), and the network as a whole is a superposition of states with a particular probability function (qua wave-function).

Now consider the Everett/Many-Worlds Interpretation (MWI). As a rough first approximation, this interpretation holds quantum superposition is to be understood in terms of *equally real quantum worlds*. Rather than positing an indeterministic wave function collapse, the MWI holds that measurement causes quantum systems to *decohere* into many different worlds in a *deterministic manner*[107], such that there *exists* a unique world for each possible state of a quantum system given by the wave-function (and so, an uncountably infinite number of parallel universes[108]). Although the MWI has been argued to resolve various problems with the Copenhagen interpretation (including the measurement problem, though this remains debated[109]), there are many objections to it, including the obvious objections that the existence of parallel worlds is unfalsifiable and ontologically profligate, and hence pseudoscientific to posit.[110] Our new explanation of quantum mechanics suggests that we have strong explanatory grounds for positing the existence of 'many worlds'—not entire universes, but instead a vast array of *parallel interacting simulations*. Our grounds for positing the existence of these simulations is, again, that they explain features of quantum mechanics that needs explaining: why there is a quantum wave-function, why objects exist in a superposition until being measured, why quantum information travels in Gaussian wave packets, why the double-slit experiment finds what it does, and why (qua Wigner's friend) 'consciousness' (viz., rendering a player's POV) plays a privileged role in quantum mechanics, 'collapsing' the wave function so that superpositions are never observed.

To this extent, our explanation also suggests that the 'many minds interpretation' of quantum mechanics—a fringe interpretation which holds that the quantum world is comprised by many parallel *minds* projecting a 'shared physical universe'—contains some real element of truth as well. For, each 'world' in a P2P simulation is, in essence, *projected* by a measuring device (a computer processor) that is in fact outside of the 'physical world' it projects. Indeed, the P2P simulation hypothesis involves a kind of functional dualism, holding that our reality is in fact comprised by two entirely different things: (I) a 'processor' (or measuring device), that (II) reads game data (e.g., the simulation's program). Insofar as computer simulations are comprised by processors reading and projecting the physical game data to its user *as* a 4-dimensional environment (qua hologram), the POV of any user is indeed ontologically and epistemologically special: it is constituted *by a dualistic mechanism* (a processor reading game data) that *is in fact* outside of the 'physical world' it projects and measures.

We could go on—but for reasons of space, we will leave it at this. Although our model is at this point just a higher-level conceptual sketch, there are some reasons to believe that our model *may*—if future investigation can show how to realize a quantum world just like ours from P2P networking (in all of its rich details)—accomplish something that no existing interpretation of quantum mechanics promises: explaining *why* our world is quantum mechanical. Our model also suggests that extant interpretations of quantum mechanics may each be partly correct. On our model, quantum mechanics really is comprised by a superposition of states that are 'collapsed' by processes of measurement (qua Copenhagen Interpreation). But, these phenomena just are many 'worlds' (viz. the Everett Interpretation)—specifically, many parallel simulations—*measuring each other*, such that each user always experiences a near-classical environment (viz., the many-interacting worlds interpretation). Further, the explanation for how these worlds generate quantum phenomena is fundamentally relational (qua relational quantum mechanics), involving 'hidden variables' (dead reckoning algorithms), akin to Bohmian mechanics.

5. Conclusion

Critics of the simulation hypothesis have deemed it pseudoscience.[111] Others maintain that if we discovered tell-tale 'glitches in the Matrix', then that might be physical evidence that we live in a simulation.[112] If this article is correct, then quantum mechanics itself *may* be such 'glitches', constituting real evidence, by inference to the best explanation, that we live in a massive peer-to-peer network of interacting simulations using dead reckoning solutions to ensure an otherwise 'believable' world for observers.

Declarations

Ethical Approval

Not applicable—no research on human or animal subjects was completed.

Competing Interests

The author has no competing financial or personal interests.

Authors' contributions

Not applicable. The author is the sole contributor.

Funding

Not applicable. No funding was received for this project.

Availability of data and materials

Not applicable. No datasets were used.

References

Adler, Stephen L. (2003). Why decoherence has not solved the measurement problem: a response to P.W. Anderson. *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics* 34(1): 135–142

Anikin, Anatoly Yu, Dobrokhotov, Sergey Yu, Klevin, Alexander I., & Tirozzi, Brunello (2019). Short-Wave Asymptotics for Gaussian Beams and Packets and Scalarization of Equations in Plasma Physics. *Physics 1*(2): 301–320.

Arvan, Marcus (2013). A New Theory of Free Will. *Philosophical Forum* 44 (1):1-48.

Arvan, Marcus (2014). A Unified Explanation of Quantum Phenomena? The Case for the Peer-to-Peer Simulation Hypothesis as an Interdisciplinary Research Program. *Philosophical Forum* 45(4):433-446.

Arvan, Marcus (2015). The Peer-to-Peer Simulation Hypothesis and a New Theory of Free Will. *Scientia Salon*.

Aviation (2017). How heading bug is calculated during cross wind and tail wind?., https://aviation.stackexchange.com/questions/23202/how-heading-bug-is-calculated-during-cross-wind-and-tail-wind.

Ballentine, L. E. (1970). The Statistical Interpretation of Quantum Mechanics. *Reviews of Modern Physics* 42(4): 358–381.

Beane, S. R., Davoudi, Z., & Savage, M. J. (2012). Constraints on the Universe as a Numerical Simulation. *arXiv preprint arXiv:1210.1847*.

Bell, John (1990). Against 'measurement'. *Physics World* 3(8): 33–41.

Bennett, Jay (2017). How Many Particles Are in the Observable Universe?, *Popular Mechanics*, https://www.popularmechanics.com/space/a27259/how-many-particles-are-in-the-entire-universe/.

Bhattacharjee, Deep (2021). Mandela effect & Déjà vu: Are we living in a simulated reality?. https://www.techrxiv.org/ndownloader/files/30884881/1.

Bhaumik, Mani L. (2017). Is Schrödinger's Cat Alive?. *Quanta 6*(1): 70–80.

Bohm, David (1952). A Suggested Interpretation of the Quantum Theory in Terms of 'Hidden Variables. I. *Physical Review* 85(2): 166–179.

Bostrom, Nick (2003). Are we living in a computer simulation? *Philosophical Quarterly* 53 (211): 243–255.

Bostrom, Nick (2013). Simulation Hypothesis: Will We Find Glitches, https://www.youtube.com/watch?v=Gx7nuThRxgE.

Brown, Harvey R. & Wallace, David (2005). Solving the measurement problem: de Broglie–Bohm loses out to Everett. *Foundations of Physics* 35(4): 517–540.

Campbell, Tom; Owhadi, Houman; Sauvageau, and David Watkinson (2017). On Testing the Simulation Theory. *International Journal of Quantum Foundations* 3: 78–99.

Chalmers, David (2022). *Reality+: Virtual Worlds and the Problems of Philosophy*. New York: W.W. Norton & Co.

Clark, Josh (n.d.). How Quantum Suicide Works: The Copenhagen Interpretation. https://science.howstuffworks.com/innovation/science-questions/quantum-suicide4.htm.

Clegg, Brian (2013). 20 amazing facts about the human body. *The Guardian,* https://www.theguardian.com/science/2013/jan/27/20-human-body-facts-science.

Cloudfare (n.d.). What is a packet?, https://www.cloudflare.com/learning/network-layer/what-is-a-packet/.

Cramer, John G. (1986). The Transactional Interpretation of Quantum Mechanics. *Reviews of Modern Physics* 58(3): 647–688.

Dobbin, Jolene (2020). Lag! Top 5 Reasons your Ping is so High. https://www.hp.com/us-en/shop/tech-takes/5-reasons-your-ping-is-so-high.

Donati, O., Missiroli, G.F., and Pozzi, G. (1973). An Experiment on Electron Interference. *American Journal of Physics* 41(5): 639–644.

Donohue, Danny (2000). Diameter, Radius of an Electron. *The Physics Factbook*, https://hypertextbook.com/facts/2000/DannyDonohue.shtml.

Eibenberger, Sandra; Gerlich, Stefan; Arndt, Markus; Mayor, Marcel, & Tüxen, Jens (2013). Matter–wave interference of particles selected from a molecular library with masses exceeding 10000 amu. *Physical Chemistry Chemical Physics 15*(35): 14696–14700.

Einstein, A. [1905] (2005). Über einen die Erzeugung und Verwandlung des Lichtes betreffenden heuristischen Gesichtspunkt. *Annalen der Physik* 14(S1 1): 164–181.

Ellis, George, & Silk, Joe (2014). Scientific method: Defend the integrity of physics. *Nature* 516(7531): 321–323.

Emery, N. (2022). Quantum correlations and the explanatory power of radical metaphysical hypotheses. *Philosophical Studies*, 1–24.

Everett, Hugh (1957). Relative State Formulation of Quantum Mechanics. *Reviews of Modern Physics* 29(3): 454–462.

Faye, Jan (2019). Copenhagen Interpretation of Quantum Mechanics. In E.N. Zalta, *The Stanford Encyclopedia of Philosophy* (Winter 2019 Edition), https://plato.stanford.edu/archives/win2019/entries/qm-copenhagen/.

Fein, Yaakov Y., Geyer, Philipp, Zwick, Patrick, Kiałka, Filip, Pedalino, Sebastian, Mayor, Marcel, ... & Arndt, Markus (2019). Quantum superposition of molecules beyond 25 kDa. *Nature Physics* 15(12): 1242–1245.

Feynman, R. P. (1948). Space-Time Approach to Non-Relativistic Quantum Mechanics. *Reviews of Modern Physics* 20(2): 367–387

Ficek, Zbigniew & Swain, Stuart (2005). *Quantum Interference and Coherence: Theory and Experiments*. New York: Springer.

Flight Literacy (2022). Dead Reckoning. https://www.flightliteracy.com/dead-reckoning/.

Fuchs, Christopher A. (2010). QBism, the perimeter of quantum Bayesianism. *arXiv preprint arXiv:1003.5209*.

Gapo, Branko (2021). What Is A Good FPS For Gaming?, https://www.gpumag.com/good-fps-for-gaming/.

Gillis, Alexander S. (2021). DEFINITION: packet loss. https://www.techtarget.com/searchnetworking/definition/packet-loss.

Goldstein, Sheldon (2021). Bohmian Mechanics. In E.N. Zalta, *The Stanford Encyclopedia of Philosophy* (Fall 2021 Edition), https://plato.stanford.edu/archives/fall2021/entries/qm-bohm/.

Grange, Eric (2016). Mesh World P2P Hypothesis, https://www.delphitools.info/DWSH/.

Griffiths, Robert B. (1984). Consistent histories and the interpretation of quantum mechanics. *Journal of Statistical Physics* 36(1–2): 219–272

Hall, Michael J., Deckert, Dirk-André, & Wiseman, Howard M. (2014). Quantum phenomena modeled by interactions between many classical worlds. *Physical Review X* 4(4): 041013.

Heisenberg, W. (1927), Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik. *Zeitschrift für Physik* 43(3–4): 172–198

Hossenfelder, Sabine (2021). Why the simulation hypothesis is pseudoscience. *Big Think*, https://bigthink.com/thinking/why-the-simulation-hypothesis-is-pseudoscience/.

Knutsson, Björn, Lu, Honghui, Xu, Wei, & Hopkins, Bryan (2004). Peer-to-peer support for massively multiplayer games. *IEEE INFOCOM 2004* (Vol. 1). IEEE, https://ieeexplore.ieee.org/abstract/document/1354485.

Lengyel, Eric (2012). *Mathematics for 3D Game Programming And Computer Graphics*. Third Edition. Charles River Media.

Li, Juncheng & Chen, Sheng (2016). The cubic α-Catmull-Rom spline. *Mathematical and Computational Applications* 21(3), 33: 1–14.

Liu, Xiaokang, Grassi, Flavia, Spadacini, Giordano, & Pignari, Sergio A. (2019). Physically based modeling of hand-assembled wire bundles for accurate EMC prediction. *IEEE Transactions on Electromagnetic Compatibility* 62(3): 914–922.

Lockwood, M. (1996). 'Many Minds'. Interpretations of Quantum Mechanics. *The British journal for the philosophy of science* 47(2): 159–188.

Lyon, Aidan (2020). Why are normal distributions normal?. *The British Journal for the Philosophy of Science* 65: 621–649.

Merriam-Webster (2022). Eigenstate. *Merriam-Webster.com Dictionary*, https://www.merriam-webster.com/dictionary/eigenstate. Accessed 29 Sep. 2022.

Moravec, Hans (1998). Simulation, Consciousness, and Existence, https://www.frc.ri.cmu.edu/~hpm/project.archive/general.articles/1998/SimConEx.98.html.

Murphy, Curtiss (2011). Believable Dead Reckoning for Networked Games. In Eric Lengyel (ed.), *Game Engine Gems 2*, pp. 308–26. Natick: AK Peters Ltd.

Osnaghi, Stefano; Freitas, Fabio; Olival Freire, Jr (2009). The Origin of the Everettian Heresy. *Studies in History and Philosophy of Modern Physics* 40(2): 97–123.

Pais, A. (1979). Einstein and the Quantum Theory. *Reviews of Modern Physics* 51: 863-914.

Planck, M. (1901). Ueber die Elementarquanta der Materie und der Elektricität. *Annalen der Physik* 309(3): 564–566.

Proietti, Massimiliano, Pickston, Alexander, Graffitti, Francesco; Barrow, Peter; Kundys, Dmytro; Branciard, Cyril, Ringbauer, Martin, & Fedrizzi, Alessandro (2019). Experimental test of local observer independence. *Science Advances*, *5*(9), eaaw9832.

Roor, Mishal (2020). 7 Advantages and Disadvantages of Peer to Peer Network. https://www.hitechwhizz.com/2020/11/7-advantages-and-disadvantages-drawbacks-benefits-of-p2p-network.html.

Rovelli, Carlo (1996). Relational quantum mechanics. *International Journal of Theoretical Physics*, *35*(8), 1637-1678.

Rudyb (2020). Navigation. 05 Wind Correction Angle, https://flightsimnavigation.wordpress.com/2020/04/07/05-wind-correction-angle/

Schaff, J.-F., Langen, T., & Schmiedmayer, J. (2014). Interferometry with atoms. *La Rivista del Nuovo Cimento*, *37*(10), 509–589.

Schlosshauer, Maximilian (2005). Decoherence, the measurement problem, and interpretations of quantum mechanics. *Reviews of Modern Physics* 76 (4): 1267–1305

Schrödinger, Erwin (1935). Die gegenwärtige Situation in der Quantenmechanik. *Naturwissenschaften* 23(48): 807–812.

Sebens, Charles T. (2015). Quantum Mechanics as Classical Physics. *Philosophy of Science* 82 (2). Cambridge University Press: 266–91. doi:10.1086/680190

Sivasundaram, Sujeevan & Nielsen, Kristian Hvidtfelt (2016). Surveying the attitudes of physicists concerning foundational issues of quantum mechanics. *arXiv preprint arXiv:1612.00676*.

Stanford, Kyle (2021). Underdetermination of Scientific Theory. In E.N. Zalta, *The Stanford Encyclopedia of Philosophy* (Winter 2021 Edition), https://plato.stanford.edu/archives/win2021/entries/scientific-underdetermination/.

Turchin, Alexey & Yampolskiy, Roman (2019). Glitch in the Matrix: Urban Legend or Evidence of the Simulation?, https://philarchive.org/archive/TURGITv1.

Vaidman, Lev (2021). Many-Worlds Interpretation of Quantum Mechanics. In E.N. Zalta, *The Stanford Encyclopedia of Philosophy* (Fall 2021 Edition), https://plato.stanford.edu/archives/fall2021/entries/qm-manyworlds/.

Van Verth, James and Bishop, Lars (2008). *Essential Mathematics in Games and Interactive Applications: A Programmer's Guide*, 2nd edition. Morgan Kaufmann.

Velimirovic, Andreja (2021). What is a Dedicated Server for Gaming? And Why You Need It. https://phoenixnap.com/blog/what-is-a-dedicated-server-for-gaming.

Wallace, David (2010). Decoherence and Ontology, or: How I Learned To Stop Worrying And Love FAPP. In Simon Saunders, Jonathan Barrett, Adrian Kent, & David Wallace (eds.). *Many Worlds? Everett, Quantum Theory and Reality*, pp. 53–72. Oxford University Press.

Walker, T. (2021). *Dead Reckoning for Distributed Network Online Games*. Master's thesis, University of Waterloo.

Wang, Jiachen; Zhang, Tianyu; Shen, Dawei; Hu, Xiaobo Sharon, & Han, Song (2021). Apas: An adaptive partition-based scheduling framework for 6tisch networks. In 2021 IEEE 27th Real-Time and Embedded Technology and Applications Symposium (RTAS): 320–332.

Whitworth, Brian (2008). The Physical World as a Virtual Reality. https://arxiv.org/abs/0801.0337.

Wigner, Eugene P. (1961). Remarks on the Mind-Body Question. In Good, I. J. (ed.). *The Scientist Speculates: An Anthology of Partly-Baked Ideas*. London: Heinemann.

Wittgenstein, Ludwig (1922). *Tractatus Logico-Philosophicus*. Charles Kay Ogden (trans.), London: Routledge & Kegan Paul Ltd.

Young, Thomas (1804). The Bakerian lecture. Experiments and calculation relative to physical optics. *Philosophical Transactions of the Royal Society of London* 94: 1–16

Zeh, H. Dieter (1970). On the Interpretation of Measurement in Quantum Theory. *Foundations of Physics*, vol. 1, pp. 69–76.

Zurek, Wojciech (2009). Quantum Darwinism. *Nature Physics* 5(3): 181–188.

Notes

[1] Wittgenstein (1921), §6.371.

[2] Faye (2019).

[3] Vaidman (2021).

[4] Goldstein (2021).

[5] Alternative interpretations include relational quantum mechanics (Rovelli 1996), the consistent histories interpretation (Griffiths 1984), ensemble interpretation (Ballentine 1970), Quantum Bayesianism (Fuchs 2010), transactional interpretation (Cramer 1986), and Many-Interacting Worlds interpretation (Hall et al. 2014).

[6] For objections to the Copenhagen Interpretation, see e.g., Bell (1990); for the Many Worlds Interpretation, see Ellis & Silk (2014); and for Bohmian Mechanics, see Brown & Wallace (2015). More generally, interpretations of quantum mechanics run up against the problem of underdetermination of theory by observation (Stanford 2021), as each interpretation purports to be consistent with all observations of quantum phenomena.

[7] Arvan (2014), pp. 434–4.

[8] See Clark (n.d.) for an informal explanation. For a more technical explanation, see Faye (2019).

[9] Everett (1957), Wallace (2010).

[10] Zurek (2009).

[11] Bohm (1952).

[12] Arvan (2014).

[13] Flight Literacy (2022).

[14] Murphy (2011), Walker (2021).

[15] Lengyel (2012), Van Verth & Bishop (2018)

[16] Wigner (1961).

[17] Bostrom (2013), Turchin & Yampolskiy (2019), Bhattacharjee (2021).

[18] [Acknowledgments redacted for review].

[19] Hall et al. (2014); Sebens (2015).

[20] Bostrom (2003), Arvan (2014), Chalmers (2022), Emery (2022).

[21] Beane et al (2012) and Campbell et al. (2017).

[22] Moravec (1998), Whitworth (2008).

[23] Grange (2016).

[24] Emery (2022).

[25] Arvan (2013, 2014, 2015).

[26] Velimirovic (2021).

[27] Ibid.

[28] Ibid.

[29] Ibid.

[30] As Campbell et al. (2017) point out, many games only 'render' content for players in their local environment, or player's point-of-view (POV), leaving other parts of the simulated environment *unrendered* until observed by some player(s). We will return to this in §3.3, as it will play a role in our model's explanation of the double-slit experiment.

[31] Knutsson et al. (2004). As Grange (2016) notes, an efficient P2P simulation requires an *adaptive space partition mesh*, or algorithms that dynamically allocate computing resources to entire groups of computers (see Wang et al. 2021). To simplify greatly, an adaptive partition mesh assigns fewer nodes to process informationally 'thin' parts of a simulation (e.g. empty space in voids), and more nodes to informationally rich areas (such as planets with many players, flora and fauna, and complex environments), so that no single computer's processing power is exceeded and all players experience the game without 'slowdown.'

[32] Knuttson et al (2021), p. 5.

[33] Ibid.

[34] Roor (2020).

[35] Walker (2021), pp. 1-2.

[36] Ibid., p. 2.

[37] Merriam-Webster (2022).

[38] "The Archer - Basic Stick Figure Pose" by Iohannes Crispien II is marked with Public Domain Mark 1.0. "File:Cartesian planes style.png" by Ali Mohamed Tarek is licensed under CC BY-SA 4.0. Transformations by Author are licensed under CC BY-SA 4.0.

[39] "The Archer - Basic Stick Figure Pose" by Iohannes Crispien II is marked with Public Domain Mark 1.0. "File:Cartesian planes style.png" by Ali Mohamed Tarek is licensed under CC BY-SA 4.0. Transformations by Author are licensed under CC BY-SA 4.0.

[40] Lyon (2020), p. 622.

[41] "Standard Normal Distribution" by D Wells is licensed under CC BY-SA 4.0.

[42] Lyon (2020), p. 622.

[43] Walker (2021), pp. 1-2.

[44] "File:Normal Distribution PDF.svg" by Inductiveload is released into the public domain.

[45] Schaff et al. (2014), p. 12, Figure 1.

[46] "File:QHO-coherent3-phasesqueezed2dB-animation.gif" by Geek3 is licensed under CC BY 3.0. Snapshots were made.

[47] "File:Indeterminacy principle.gif" by Thierry Dugnolle is marked with CC0 1.0.

[48] "File:Mplwp gaussian wavepacket f4.svg" by Geek3 is licensed under CC BY 3.0. Envelope packet drawn by Author.

[49] "File:Rotations on the complex plane.svg" by Keφr is marked with CC0 1.0. "Wavepacket-a2k4-en" by Xcodexif licensed under CC BY-SA 4.0. Wavepacket snapshot licensed by Author under CC BY-SA 4.0.

[50] Ficek and Swain (2005).

[51] "File:ConstructiveAndDestructiveInterference.png" by KaWus1093 is licensed under CC BY-SA 3.0.

[52] "File:QuantumHarmonicOscillatorAnimation.gif" by Sbyrnes321 is marked with CC0 1.0.

[53] Gapo (2021).

[54] Walker (2021), p. 2.

[55] Dobbin (2020).

[56] Gillis (2021).

[57] See e.g., Murphy (2011), Walker (2021).

[58] Walker (2021), p. 2, Figure 1.2. Annotations added.

[59] "File:Navigation - dead reckoning through a current.svg" by Korektor is licensed under CC BY-SA 3.0. Deduced course correction added and l

[60] Rudyb (2020), Aviation (2017).

[61] Walker (2021), p. 10, Figure 3.1.

[62] "File:Wind drift.png" by Abuk Sabuk is licensed under CC Attribution-Share Alike 3.0 Unported license. Rotations and other modifications added by Author under CC Attribution-Share Alike 3.0 Unported/

[63] "File:Wave characteristics.svg" by Krishnavedala is made available under the Creative Commons CC0 1.0 Universal Public Domain Dedication.

[64] Figure is from Walker (2021), p. 23.

[65] See Lengyel (2004), pp. viii, 320–338, 537–539.

[66] "File:Bezier curve.svg" by Marian Sigler is released by the copyright holder into the public domain. "File:Catmull-Rom Spline.png" by Hadunsford is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license. "File:Hermite Basis.svg" by Олюсь is released into the public domain.

[67] Murphy (2011), §1.3; Walker (2021), §§3.4–3.5 and §5.

[68] Li and Chen (2016), p. 8. No special permission is required.

[69] Ibid., p. 12, Figure 7.

[70] "File:Quantum mechanics travelling wavefunctions wavelength.svg" by Maschen is released by the copyright holder into the public domain.

[71] "File:Electromagnetic wave2.svg" by Francois~frwiki is licensed under the Creative Commons Attribution-Share Alike 4.0 International license.

[72] Cloudflare (n.d.)

[73] Planck (1901), Einstein [1905].

[74] Young (1804).

[75] "File:Two-Slit Experiment Particles.svg" by inductiveload is in the public domain.

[76] "File:Two-Slit Experiment Light.svg" and "File:Two-Slit Experiment Electrons.svg" by inductiveload are both in the public domain.

[77] Donati et al. (1973); Eibenberger et al. (2013); and Fein et al. (2019).

[78] Pais (1979), p. 907.

[79] Campbell et al. (2017), p. 3.

[80] Image snapshots taken from: https://www.reddit.com/r/interestingasfuck/comments/7v95as/how_games_render_as_you_move_the_camera/

[81] Clegg (2013).

[82] Bennett (2017).

[83] Campbell et al. (2017), p. 3.

[84] Donoghue (2000).

[85] Feynmann (1948).

[86] "File:Feynman paths.png" by Sachin48 sps is licensed under the Creative Commons Attribution-Share Alike 4.0 International license.

[87] Modifications by Author to File:Two-Slit Experiment Electrons.svg" by inductiveload and "File:Feynman paths.png" by Sachin48 sps are licensed under the Creative Commons Attribution-Share Alike 4.0 International license.

[88] Modifications by Author to "File:Two-Slit Experiment Electrons.svg" by inductiveload and "File:Catmull-Rom Spline.png" by Hadunsford is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license

[89] Figure based on Murphy (2011), p. 5, Figure 1.4.

[90] Walker (2021), pp. 13-15.

[91] Image: Murphy (2011), p. 4, Figure 1.3.

[92] Based on Murphy (2011), p. 5, Figure 1.4.

[93] Walker (2021), Chapter 5.

[94] Murphy (2011), p. 29; italics added.

[95] Walker (2021), p. 28, Figure 5.1.

[96] Walker (2021), p. 35, Figure 5.4.

[97] Ibid., p. 37, Figure 5.5.

[98] Wigner (1935).

[99] Schrödinger (1935).

[100] Sivasundaram & Nielsen (2016), p. 11.

[101] Bhaumik (2017), §6.

[102] Proietti et al. (2019).

[103] "File:Stickman icon.svg" is licensed under the Creative Commons Attribution-Share Alike 2.0 Generic license. "The Archer - Basic Stick Figure Pose" by Iohannes Crispien II is marked with Public Domain Mark 1.0. Modifications are made by Author under Attribution-Share Alike 2.0 Generic license.

[104] Faye (2019)

[105] Heisenberg (1927).

[106] Bell (1990).

[107] Zeh (1970); Schlosshauer (2005).

[108] Osnaghi et al. (2009).

[109] Adler (2003).

[110] Ellis & Silk (2014).

[111] Hossenfelder (2021).

[112] Bostrom (2013), Turchin & Yampolskiy (2019), Bhattacharjee (2021).