A Comprehensive Analysis of Probabilistic Event-Time Geometry: From Formalism to Application in Neuromorphic and Distributed Systems

Foundations of Deterministic Event-Time Geometry (D-ETG)

The analysis of complex, event-driven systems, ranging from distributed computing networks to neuromorphic hardware, necessitates a rigorous framework for reasoning about causality and time. The classical approach, which may be termed Deterministic Event-Time Geometry (D-ETG), provides an elegant and powerful formalism for this purpose. It establishes an idealized model of causality by drawing a direct and fruitful analogy to the geometry of spacetime in special relativity. This deterministic foundation, while powerful in its clarity, operates under assumptions of perfect knowledge and predictability, thereby setting a crucial baseline against which the complexities of real-world uncertainty can be measured and modeled. By defining the core components of D-ETG—the event, the causal order, and the invariant interval—we can establish a clear understanding of its strengths as a theoretical tool and its inherent limitations in the face of stochastic phenomena.

The Event as a Point in Spacetime

In the deterministic formalism, the fundamental ontological unit is the event. An event, denoted as e, is conceptualized as a perfectly localized point in a composite space-time manifold, defined by a specific spatial location l and a precise timestamp t. Thus, an event is a tuple e = (l, t). The set of all possible locations, L, is endowed with a distance function, d(l_1, l_2), which quantifies the spatial separation or, more abstractly, the minimum communication latency between any two locations. The time domain, T, is typically represented by the set of real numbers, \mathbb{R}, ensuring that every event is assigned a real-valued time coordinate. For instance, in a distributed system, l might be the unique identifier of a processor or server, while t represents the value of its local clock at the moment a message is sent or a transaction is committed. Similarly, in a neuromorphic system, l could identify a specific artificial neuron, and t would be the exact time it generates a spike. This conceptualization establishes a world of discrete, point-like occurrences whose positions in both space and time are known with absolute certainty.

The Geometry of Causality

Upon this foundation of precisely defined events, D-ETG imposes a strict partial order governed by the principle of causality. This principle is not merely logical but is grounded in the physical or systemic constraints of information propagation. An event e_1 = (l_1, t_1) can causally influence

a subsequent event $e_2 = (l_2, t_2)$, a relationship denoted as $e_1 \to e_2$, only if there is sufficient time for a signal to travel from location l_1 to l_2. This is formalized by a fundamental causal delay constraint. If we define a maximum signal propagation speed within the system as c, then a necessary condition for $e_1 \to e_2$ is given by the inequality:
This inequality asserts that the temporal separation between the two events, $\Delta t = t_2 - t_1$, must be at least as great as the minimum time required to traverse the spatial distance, $\Delta x = d(l_1, l_2)$, at the maximum speed c. If this condition is violated, e_1 is causally prohibited from influencing e_2. This formulation creates a strict, binary partial order on the set of all events.
This geometric view of causality represents a significant conceptual advance over purely logical or topological frameworks for event ordering, such as Lamport's "happened-before" relation. Lamport's model defines a partial order based on the sequence of events within a single process and the sending and receiving of messages between processes. It can definitively answer the question, "Did event A happen before event B?" based on this logical chain of dependencies. However, it lacks any notion of metric distance or real time. It cannot answer questions like, "By how much time did A precede B?" or "Could A still have caused B if the network latency were higher?". D-ETG, by introducing the metric components $d(l_1, l_2)$ and c, grounds the causal relationship in the physical reality of the system. The causal inequality is a physical or systemic constraint, not just a logical one. This elevation from a logical to a geometric framework introduces the ability to perform quantitative analysis of system properties like latency, temporal slack, and performance budgets, which is a prerequisite for the rigorous design of systems with hard real-time constraints where deadlines are absolute and must be mathematically guaranteed.

The Minkowski-like Interval

To quantify the separation between events in a way that respects the underlying causal structure, D-ETG introduces an invariant measure analogous to the spacetime interval in Minkowski space. Given two events, $e_1 = (l_1, t_1)$ and $e_2 = (l_2, t_2)$, with temporal separation $\Delta t = t_2 - t_1$ and spatial separation $\Delta x = d(l_1, l_2)$, the squared event-time interval is defined as:
This single scalar quantity provides a powerful, tripartite classification of the relationship between any pair of events, directly reflecting their causal connectability.

- **Timelike Separation ($(\Delta s)^2 > 0$):** When the temporal separation is large enough to overcome the spatial distance, the events are causally connectable. Assuming $\Delta t > 0$, event e_2 lies within the future causal cone of e_1, meaning e_1 could have influenced e_2. A large positive value for $(\Delta s)^2$ indicates that there was ample time for a signal to propagate between the locations.
- **Spacelike Separation ($(\Delta s)^2 < 0$):** When the spatial distance is too great for the given temporal separation, the events are causally disconnected. No signal traveling at or below speed c could have connected them. They exist outside of each other's causal cones.
- **Lightlike Separation ($(\Delta s)^2 = 0$):** This boundary case represents events connected by a signal traveling at precisely the maximum speed c. Event e_2 lies on the edge of e_1's future causal cone.

This classification is absolute and exhaustive, partitioning all event pairs based on their intrinsic causal relationship. The causal cone of an event e is thus defined as the set of all points in space-time that have a timelike or lightlike separation from e, cleanly delineating the region of potential causal influence from the region of causal independence.

The Event-Time Invariance Theorem

The theoretical cornerstone of D-ETG is the Event-Time Invariance Theorem, which states that the value of the squared event-time interval, $(\Delta s)^2$, is invariant under a specific class of coordinate transformations known as Lorentz-like transformations. These transformations model the relationship between different "inertial frames of reference." In the context of distributed systems, a frame can be thought of as a particular clock synchronization scheme or the perspective of a specific observer or subsystem.

The theorem guarantees that while different observers (i.e., different frames) may disagree on the individual time and space coordinates of events—a phenomenon analogous to the relativity of simultaneity in physics—they will all calculate the exact same value for the interval $(\Delta s)^2$ between any two events. The profound implication of this invariance is that the causal structure of the system is absolute. All observers, regardless of their reference frame, will agree on the classification of an event pair as timelike, spacelike, or lightlike. The question of whether two events *can* be causally related is an objective, frame-independent truth.

This deterministic nature is simultaneously D-ETG's greatest strength and its most significant weakness in practical application. It provides a powerful and consistent framework for analyzing systems where behavior is perfectly predictable, such as theoretical models of computation or fully synchronous digital circuits. However, its core assumptions—perfectly accurate clocks, zero transmission jitter, and constant, known propagation delays—are systematically and often dramatically violated in the asynchronous, noisy, and complex environments of modern distributed systems and neuromorphic hardware. This inherent "brittleness" and the discrepancy between its idealized model and the stochastic reality of physical systems directly motivate and necessitate the development of a probabilistic extension.

The Probabilistic Extension: Formalizing P-ETG

The deterministic framework of Event-Time Geometry, while providing a clear and powerful model of causality, rests on an idealization of perfect certainty that is seldom met in practice. Real-world systems are replete with sources of uncertainty: clocks drift, network packets experience variable delays (jitter), sensor measurements are noisy, and processing times fluctuate. To create a formalism that can accurately model and reason about such systems, it is necessary to extend D-ETG into the probabilistic domain. Probabilistic Event-Time Geometry (P-ETG) achieves this by systematically replacing each deterministic concept with a stochastic counterpart. This transformation moves from a world of precise points and sharp boundaries to one of probability distributions and graded confidence, yielding a richer, more nuanced, and ultimately more realistic model of causality under uncertainty.

From Points to Clouds: Representing Probabilistic Events

The most fundamental shift in P-ETG is the redefinition of the event itself. An event is no longer a deterministic point (l, t) with precise coordinates but is instead represented as a random variable, $E = (L_E, T_E)$, characterized by a joint probability distribution, $P_E(l, t)$, over the space-time manifold. This distribution can be visualized as a "probability cloud" or "blob" in a space-time diagram, where the density of the cloud at any point (l, t) corresponds to the

likelihood that the event occurred at that specific location and time.
This representation directly captures the various sources of real-world uncertainty. For example, the timestamp of a neuron's spike in a neuromorphic system might be modeled as a Gaussian random variable, $T_E \sim \mathcal{N}(\mu_t, \sigma_t^2)$, where μ_t is the expected firing time and σ_t^2 is the variance due to intrinsic neural noise or clock jitter. Similarly, uncertainty in the spatial location of an event, such as a signal originating from one of several possible nodes in a wireless network, can be modeled by making L_E a random variable over the set of locations L. This "event cloud" picture acknowledges that our knowledge of an event's coordinates is often limited to a set of statistical properties (e.g., mean, covariance) or bounded intervals. The smaller the variance of the distribution, the tighter the cloud, and in the limit as the variance approaches zero, the probabilistic event converges to its deterministic point-like counterpart.

Stochastic Intervals and Fuzzy Causality

The introduction of uncertainty at the level of event coordinates has profound consequences for the entire geometric structure. Since the event coordinates (L_1, T_1) and (L_2, T_2) are now random variables, the temporal separation $\Delta T = T_2 - T_1$ and the spatial separation $\Delta X = d(L_1, L_2)$ also become random variables, each with its own distribution derived from the underlying event distributions.
Consequently, the event-time interval, now a stochastic quantity $(\Delta S)^2 = c^2(\Delta T)^2 - (\Delta X)^2$, is itself a random variable. Instead of a single, invariant scalar value, we now have a full probability distribution over possible interval values, $P((\Delta S)^2)$. This fundamentally changes the nature of causal classification. The sign of $(\Delta S)^2$, which deterministically partitions event pairs, is no longer certain. An event pair might have a 90% probability of being timelike separated (if most of the mass of the $P((\Delta S)^2)$ distribution lies above zero) and a 10% probability of being spacelike separated (if a smaller portion of the distribution falls below zero). This leads to the concept of fuzzy causal classes, where the relationship between events is described by a set of probabilities rather than a single, crisp label:

- $P_{\text{timelike}} = P((\Delta S)^2 > 0 \text{ and } \Delta T > 0)$
- $P_{\text{spacelike}} = P((\Delta S)^2 < 0)$
- $P_{\text{lightlike}} = P((\Delta S)^2 = 0)$

This probabilistic classification provides a much more nuanced and realistic depiction of causality in systems where timing is not perfectly precise.

The Probabilistic Causal Relation

With the deterministic causal condition rendered uncertain, the binary true/false nature of the causal relation $e_1 \to e_2$ is replaced by a graded measure of confidence. The causal influence probability is defined as the probability that the necessary condition for causality holds, given the distributions of the event times and locations :
This value, ranging from 0 to 1, represents our degree of belief that e_1 could have causally affected e_2. A probability near 1 indicates high confidence in a potential causal link, while a probability near 0 suggests that e_2 almost certainly occurred too soon for e_1 to have been a cause. This formalism naturally arises in distributed systems with imperfectly synchronized clocks. For example, given the known offset distributions of two servers' clocks, one can compute the probability that a log entry on server A truly occurred before a log entry on server B, even if their nominal timestamps suggest otherwise.

This concept has been formalized in recent research through relations like "likely-happened-before," which explicitly assigns a probability to the "happened-before" relation, extending Lamport's purely logical framework. This allows for a probabilistic ordering of events that would be considered concurrent in a deterministic view. The trade-off is that this probabilistic relation may not be transitive; it is possible that $P(A \to B)$ and $P(B \to C)$ are both high, while $P(A \to C)$ is not. As a result, the causal structure of the system is no longer a strict partial order but is more accurately represented as a weighted directed acyclic graph, where the edges are annotated with these causal probabilities.

Fuzzy Lightcones and Stochastic Invariance

The sharp, well-defined boundary of the deterministic causal cone, defined by $(\Delta s)^2 = 0$, dissolves under uncertainty into a "fuzzy" or blurred transition region. We can visualize this as a cone whose edges are smeared out, reflecting the probabilistic nature of causality at the boundary. An event that would be classically just outside the lightcone (spacelike) might now have a small but non-zero probability of being influenced, due to the possibility of timing fluctuations. Conversely, an event just inside the classical cone might, in some realizations, occur too early to be influenced.

This fuzzy lightcone can be formalized by defining probabilistic causal cones based on confidence levels. For example, the 90% causal cone of an event e_1 could be defined as the set of all space-time points (l, t) for which the causal influence probability $P(e_1 \to (l,t))$ is greater than or equal to 0.9. The boundary of this region would be a level set of the influence probability function.

Just as invariance is the bedrock of D-ETG, a corresponding principle of stochastic invariance is crucial for P-ETG. Since the interval $(\Delta S)^2$ is now a random variable, we can no longer require that its specific value be preserved across reference frames. Instead, stochastic Lorentz invariance demands that the *statistical law*, or the full probability distribution, of the interval is invariant. This ensures that any physically meaningful probabilistic statement about the system is frame-independent. For example, the value of $P(e_1 \to e_2)$ must be the same for all observers. If one observer calculates a 95% chance of a causal link, all other observers in different inertial frames must arrive at the same conclusion. This principle acts as a critical consistency check, ensuring that the introduction of probability does not violate the fundamental symmetries of the underlying geometry.

This transition from a deterministic to a probabilistic framework can be likened to the historical evolution in physics from classical mechanics to quantum and statistical mechanics. It represents a fundamental shift from a "God's-eye view" of a system, where all state variables are perfectly known, to an observer-centric perspective, where knowledge is fundamentally incomplete and must be described by probability distributions. The "fuzzy lightcone" is conceptually analogous to quantum phenomena like tunneling, where a particle has a non-zero probability of existing in a classically forbidden state. In P-ETG, an event has a non-zero probability of exerting causal influence even if its mean coordinates would place it in a classically forbidden (spacelike) region. This reframes P-ETG not merely as an engineering tool for handling noise, but as a deeper, more physically realistic model of information and causality.

Consequently, the notion of "correctness" in system design is transformed. In D-ETG, correctness is absolute: a deadline is either met or missed. In P-ETG, correctness becomes a probabilistic guarantee: for example, "the probability of a deadline miss is less than 10^{-9}." This paradigm shift compels system designers to engage in explicit risk management, quantitatively balancing performance gains against the calculable probability of failure, a concept that is central to modern fields like probabilistic scheduling.

Feature	Deterministic ETG (D-ETG)	Probabilistic ETG (P-ETG)	Source
Event Representation	A point (I, t) with precise coordinates.	A probability distribution P(I, t) ("event cloud").	
Causal Relation	Binary: $e_1 \to e_2$ is either true or false.	Probabilistic: A graded confidence $P(e_1 \to e_2) \in$.	
Event-Time Interval	A single scalar invariant $(\Delta s)^2$.	A random variable with a distribution $P((\Delta S)^2)$.	
Causal Cone	Sharp, well-defined boundary defined by $(\Delta s)^2 = 0$.	Fuzzy boundary, defined by probability level sets.	
Invariance	The scalar value of $(\Delta s)^2$ is invariant across frames.	The statistical distribution of $(\Delta S)^2$ is invariant.	

Advanced Structures and Mathematical Connections

The formalism of Probabilistic Event-Time Geometry is not an isolated or ad-hoc construction for handling noise. Rather, it is deeply intertwined with several powerful and established mathematical frameworks. Recognizing these connections is crucial, as it allows for the importation of sophisticated tools and theoretical insights from fields such as information geometry, stochastic differential geometry, and the theory of probabilistic graphical models. These connections provide P-ETG with a rigorous mathematical underpinning, demonstrating that it is a natural synthesis of geometry, probability, and information theory. This synthesis enables a more profound understanding of the structure of causality under uncertainty and provides a rich arsenal of computational techniques for analysis and inference.

An Information-Geometric Perspective

When events are elevated from points to probability distributions, the simple Euclidean or Minkowski distance is no longer sufficient to quantify the separation between them. The natural mathematical language for measuring distances and defining geometry on a space of probability distributions is Information Geometry. This field treats a family of probability distributions as points on a differential manifold, the "statistical manifold," and equips it with a metric that captures the notion of distinguishability between distributions.

Several metrics from information geometry are relevant to P-ETG:

- **Fisher-Rao Distance:** The Fisher information metric defines an infinitesimal distance between two nearby distributions based on how sensitively the log-likelihood of the distribution changes with respect to its parameters. The integrated distance along a path on the manifold, known as the Fisher-Rao distance, provides a measure of how statistically distinguishable two event-clouds are. For example, if two spikes in an SNN are represented by Gaussian distributions over time, their Fisher-Rao distance quantifies how easily a downstream neuron could tell their timings apart. This has direct practical implications for analyzing the capacity and robustness of temporal codes in the presence of jitter. A key property of this metric is its invariance to reparameterization of the distributions, making it a fundamental measure of separation.

- **Wasserstein (Earth-Mover) Distance:** This metric offers a more intuitive notion of distance based on optimal transport theory. The Wasserstein distance between two event distributions $P_1(I, t)$ and $P_2(I, t)$ is defined as the minimum "cost" required to transport the probability mass of P_1 to match the distribution of P_2, where the cost is related to the underlying distance in the space-time manifold. If two event clouds are far apart in mean time but have significant overlap due to large variances, their Wasserstein distance will be smaller than the simple distance between their means, correctly capturing their statistical proximity.
- **Kullback-Leibler (KL) Divergence:** While not a true metric due to its asymmetry, the KL divergence, $D_{KL}(P_1 || P_2)$, quantifies the information gain when one revises beliefs from a prior distribution P_2 to a posterior distribution P_1. It can be used to measure the "surprise" or difference between two event distributions and is central to Bayesian inference and variational methods.

The connection to information geometry reveals a profound unification: P-ETG is the domain where the Minkowski-like geometry of causality, probability theory, and information theory intersect. The Fisher Information Metric, for instance, is simultaneously a geometric concept (defining a Riemannian metric on the statistical manifold) and an information-theoretic one (via the Cramér-Rao bound, it sets a lower limit on the variance of any unbiased estimator of the distribution's parameters). This synthesis suggests that the very structure of uncertainty in event-based systems has its own intrinsic geometry, which can be explored and exploited.

Modeling Dynamic Uncertainty with Stochastic Differential Geometry

Many sources of uncertainty in real-world systems are not static but evolve dynamically over time. A prime example is clock drift in distributed systems, where the offset of a local clock relative to a reference standard changes continuously and randomly. Stochastic Differential Geometry (SDG) provides the mathematical framework for modeling such dynamic processes on manifolds.

Within this framework, the coordinates of an event, or related parameters like clock offset, can be modeled as solutions to a stochastic differential equation (SDE). For instance, the time coordinate T_{node} of a local clock might be modeled as a diffusion process that drifts away from an ideal reference time t:

Here, α_t could be a slowly varying random process representing the clock's frequency drift (skew), and dW_t is a Wiener process (Brownian motion) term representing random, unpredictable jitter. By solving such SDEs, one can derive the probability distribution of the clock's offset at any future time, often yielding a distribution (like a Gaussian) whose variance grows over time. This allows for a principled analysis of how uncertainty accumulates along long causal chains or over long system runtimes. SDG provides the tools to study how these stochastic trajectories interact with the geometric boundaries of the system, such as the causal cones, enabling the calculation of probabilities like "what is the chance that this sequence of causally dependent tasks will miss its cumulative deadline due to accumulating jitter?".

P-ETG as a Specialized Temporal Probabilistic Graphical Model

Probabilistic Graphical Models (PGMs) are a cornerstone of modern artificial intelligence and machine learning for representing and reasoning about complex systems with many

interdependent random variables. P-ETG can be naturally and powerfully framed as a specific type of temporal PGM.
In this interpretation:

- **Nodes:** Each probabilistic event $E_i = (L_i, T_i)$ is a set of random variables, representing the nodes in the graph.
- **Edges:** The causal relationships between events define the directed edges of the graph. A potential causal link from E_i to E_j is represented by a directed edge $i \to j$.
- **Conditional Probabilities:** The probabilistic nature of the causal link is captured by the conditional probability distribution (CPD) associated with each node. The distribution of T_j is conditioned on the time of its causal parents, e.g., $P(T_j \mid T_i)$. The P-ETG formalism provides a strong structural prior on these CPDs: the support of the distribution for T_j is constrained by the inequality $T_j \ge T_i + d(L_i, L_j)/c$. This geometric constraint is a powerful piece of domain knowledge that is often absent in generic temporal PGMs, which may only assume temporal precedence without a metric constraint.

This connection is immensely valuable because it allows the entire arsenal of algorithms developed for PGMs to be applied to P-ETG. Standard inference techniques like belief propagation or Markov Chain Monte Carlo (MCMC) sampling can be used to answer complex probabilistic queries about the system. For example, if the timestamps of a subset of events in a network are observed (with noise), these algorithms can be used to infer the full posterior probability distribution for the timestamps of all unobserved events. Furthermore, the structure of the PGM (i.e., the causal dependencies) can itself be learned from data using structure learning algorithms, providing a method for discovering causal connectivity from observational event data.
The synthesis of these mathematical frameworks reveals that P-ETG is far more than a simple "ETG + noise" model. It is a coherent theoretical structure that unifies concepts from geometry, statistics, and computer science. This unification suggests a deep potential for P-ETG to serve as a foundational "physics" for complex information systems. Just as general relativity employs differential geometry to describe the interaction of matter and spacetime, P-ETG and its related mathematical fields may provide the language to describe the fundamental interplay of information, uncertainty, and causal structure in the large-scale computational systems of the future. Exploratory research into embedding entire "event-universes" into low-dimensional manifolds using information-geometric techniques hints at this profound possibility.

A Comparative Analysis with Adjacent Temporal Frameworks

Probabilistic Event-Time Geometry does not exist in a vacuum; it is part of a broader landscape of formalisms designed to reason about time and causality in computational systems. To fully appreciate its unique contributions, it is essential to position P-ETG relative to other well-known frameworks, including the logical clocks of Lamport and Mattern, the bounded-uncertainty model of Google's TrueTime, and the self-exciting models of spatio-temporal point processes. This comparative analysis reveals a spectrum of models, each with distinct trade-offs in terms of expressive power, computational cost, and the nature of the guarantees they provide. P-ETG emerges at the most expressive end of this spectrum, offering the richest model of uncertainty at the cost of greater complexity.

Beyond Logical Time: Lamport Clocks and Vector Clocks

The pioneering work of Leslie Lamport introduced the "happened-before" ($\to$) relation, a purely logical framework for capturing potential causality in asynchronous distributed systems. A Lamport clock is a simple counter maintained by each process, which is incremented for each local event and "piggybacked" on messages. By comparing timestamps, the system can enforce the rule that if $A \to B$, then the logical timestamp of A is less than that of B. Vector clocks extend this concept to provide a stronger guarantee: the vector timestamp of A is strictly less than that of B if and only if $A \to B$.

These logical clocks are fundamental to distributed systems theory and practice. However, their nature is topological, not metric. They establish a partial order but contain no information about real-time duration or physical separation. Two events, A and B, for which neither $A \to B$ nor $B \to A$ holds are deemed "concurrent." The system has no way to distinguish between two events that occurred nanoseconds apart and two that occurred hours apart, so long as they are not on the same causal chain.

P-ETG provides a significant extension to this model in two key ways:

1. **Metric Foundation:** By incorporating a physical distance $d(l_1, l_2)$ and a maximum signal speed c, P-ETG introduces a metric that quantifies the "gap" between events. The event-time interval $(\Delta s)^2$ is a real-valued measure of separation, not just a logical ordering.
2. **Probabilistic Ordering of Concurrent Events:** P-ETG directly addresses the ambiguity of concurrency. For two events that are concurrent under Lamport's definition, P-ETG can leverage knowledge of clock offset distributions to compute the probability that one truly occurred before the other. The "likely-happened-before" relation, $A \xrightarrow{p} B$, states that event A happened before event B with probability p. This allows the system to establish a total order with a specified degree of confidence (e.g., 99.9%), resolving ambiguity where logical clocks can only report concurrency.

Beyond Bounded Uncertainty: Google's TrueTime

Google's Spanner database introduced TrueTime, a groundbreaking approach to managing time uncertainty in a globally distributed system. TrueTime does not provide a single timestamp but rather a bounded interval, $[t_{\text{earliest}}, t_{\text{latest}}]$, which is guaranteed to contain the true, absolute time of an event. This interval-based representation allows for the implementation of externally consistent transactions. If the interval for transaction T_1 ends before the interval for transaction T_2 begins, their order is certain and globally agreed upon.

TrueTime is a powerful model for building systems that require strong correctness guarantees by reasoning about worst-case uncertainty. However, its approach is inherently conservative. If the uncertainty intervals of two events overlap, TrueTime can make no definitive statement about their order; they must be treated as potentially concurrent.

P-ETG offers a more granular and potentially more performant alternative by modeling uncertainty with a full probability distribution rather than a hard interval. While TrueTime's intervals might overlap, P-ETG can still compute the probability $P(t_A < t_B)$ by convolving the respective time distributions of events A and B. For example, even if their 99.99% confidence intervals overlap, the probability that A occurred before B might still be extremely high (e.g., 99.9999%). A system based on P-ETG can leverage this information to make more aggressive optimizations. In applications like fair scheduling, it can choose to order A before B, accepting a quantifiable and infinitesimally small risk of misordering in exchange for higher throughput or lower latency. TrueTime, by contrast, would be forced into a more conservative (and potentially slower) execution path to avoid any possibility of misordering.

Beyond Purely Temporal Excitation: Spatio-Temporal Point Processes

Temporal point processes, and particularly Hawkes processes, are a class of stochastic models used to describe the occurrence of discrete events in continuous time. A Hawkes process is characterized by its self-exciting nature: the occurrence of an event temporarily increases the rate, or conditional intensity, of future events. This makes them exceptionally well-suited for modeling phenomena with temporal clustering, such as aftershocks following an earthquake, viral posts on social media, or bursts of neural firing.

These models can be extended to the spatio-temporal domain, where each event has both a time and a location. In such models, the intensity function at a location l and time t can depend on the history of events at all other locations, typically with an influence kernel that decays with both time and spatial distance.

However, standard spatio-temporal point processes often treat space and time as separable dimensions or use a generic, isotropic distance-based kernel for spatial influence. P-ETG offers a more principled and physically motivated constraint on this influence. The core of P-ETG is the causal cone structure inherited from D-ETG. This imposes a hard, non-negotiable constraint on the influence kernel of any point process model built upon it: the influence of an event at (l_1, t_1) on any point (l_2, t_2) is strictly zero if the two points are spacelike separated. A standard Hawkes process might be parameterized to model an earthquake in California triggering one in Japan one second later (a physical impossibility), and the model would simply learn a near-zero influence coefficient. A P-ETG-based point process would assign this interaction a probability of zero *a priori*, based on the fundamental geometry of the system. This integration of a hard causal geometry provides a powerful structural prior that can improve model accuracy, reduce the number of parameters to be learned, and ensure physical plausibility. The recent emergence of research into "Geometric Hawkes Processes," which explicitly incorporate graph-based or other geometric structures to constrain the influence between processes, confirms the value of this synthesis.

The comparison of these frameworks reveals a clear hierarchy of expressive power and an associated set of engineering trade-offs. Lamport clocks are simple and computationally inexpensive but provide the weakest guarantees about real-time ordering. TrueTime is more powerful, offering real-time bounds suitable for systems demanding absolute safety, but its conservatism can lead to under-utilization of resources. P-ETG sits at the apex of this hierarchy, providing the richest possible description of uncertainty. This allows for the design of systems that make statistically optimal decisions based on expected outcomes and acceptable risk levels, but it comes at the cost of increased modeling and computational complexity. The choice of which model to use is therefore not about which is abstractly "best," but which is most appropriate for a given application's specific requirements for safety, performance, and fairness.

This calculus has profound implications, particularly for the design of decentralized systems. In domains like blockchain technology or high-frequency trading, where the precise ordering of events determines financial outcomes, traditional centralized sequencers introduce a single point of failure and arbitrary tie-breaking. A P-ETG-based approach, such as "likely-happened-before," provides a pathway toward decentralized, algorithmically fair ordering protocols, where the sequence of events is determined not by the whim of a central authority but by the verifiable statistical evidence of when those events actually occurred.

Practical Applications and Benefits in Neuromorphic Systems

Neuromorphic computing represents a paradigm shift away from the synchronous, frame-based processing of traditional von Neumann architectures towards asynchronous, event-driven computation inspired by the brain. In these systems, information is encoded not just in the rate of events (spikes) but in their precise relative timing. This temporal sensitivity makes neuromorphic hardware and the Spiking Neural Networks (SNNs) they run exquisitely powerful but also highly susceptible to the inherent noise, jitter, and variability of their physical substrate. The abstract formalism of Probabilistic Event-Time Geometry finds a direct and compelling application in this domain, providing the essential mathematical language to model, analyze, and design robust and reliable neuromorphic systems.

Modeling Reality: Quantifying Spike-Time Jitter and Delay Variability

The biological brain operates with a degree of stochasticity, and its silicon emulators are no different. The firing time of an artificial neuron is subject to "jitter"—small, random deviations from its expected firing time due to thermal noise, variations in analog circuit components, or quantization effects in digital clocks. Furthermore, the time it takes for a spike to travel from one neuron to another (axonal and synaptic delays) is not a fixed constant but a variable quantity.
P-ETG provides the natural framework for modeling this physical reality. A spike is not a deterministic event occurring at an exact time t, but rather a probabilistic event—an "event cloud"—represented by a probability distribution over time. For example, a spike's timestamp T can be modeled as a Gaussian distribution, T \sim \mathcal{N}(\mu_t, \sigma_t^2), where \mu_t is the mean firing time and \sigma_t is the standard deviation representing the magnitude of the jitter. This allows for a realistic, quantitative model of the system's underlying hardware, enabling designers to reason about the statistical behavior of their networks rather than relying on idealized, deterministic simulations.

Robustness of Temporal Codes

Many advanced computational schemes in SNNs, such as temporal coding and the formation of "polychronous groups," rely on the precise and repeatable timing of spikes to represent information. A polychronous group, for instance, is a set of neurons that can be made to fire in a specific, time-locked sequence in response to a stimulus. The reliability of such computational structures is directly threatened by spike-time jitter.
P-ETG provides the analytical tools to quantify the robustness of these temporal codes. Using the concept of causal confidence, a designer can analyze a specific polychronous chain and calculate the probability that the entire sequence of spikes will maintain its correct causal order in the presence of a given level of jitter. This is a path confidence calculation on the probabilistic event graph: the probability that spike_1 \to spike_2 \to \dots \to spike_n holds is derived from the joint probability distribution of all spike times. By performing this analysis, one can determine a critical jitter threshold beyond which the computation becomes unreliable, providing crucial guidance for hardware design. This allows for a quantitative trade-off between the precision (and thus power consumption and cost) of the hardware and the complexity of the temporal

codes it can reliably support.

Causal Inference in Spike Trains

A fundamental problem in both neuroscience and neuromorphic engineering is the inference of functional connectivity from observed activity. Given the recorded spike trains from a population of neurons, the goal is to determine which neurons are causally influencing others—that is, to reconstruct the underlying synaptic wiring diagram. Simple methods like cross-correlation can be misleading, as they can identify correlations that are not causal (e.g., due to a common, unobserved input).

P-ETG offers a more principled, geometry-aware approach to this problem. For every pair of observed spikes, one from neuron A at time t_A and one from neuron B at time t_B, one can compute the causal influence probability, $P(spike_A \to spike_B)$, taking into account the known propagation delay between the neurons and the statistical models of their jitter. By aggregating these probabilities over all spike pairs between A and B, one can compute an overall confidence score for the existence of a synaptic link $A \to B$. This method moves beyond simple statistical correlation to a measure of causality that respects the physical constraints of the system, providing a more robust method for reverse-engineering network connectivity from activity data.

Event-Based Vision and Sensor Fusion

Neuromorphic vision sensors, such as the Dynamic Vision Sensor (DVS) or event camera, are a key component of the neuromorphic ecosystem. These devices do not capture frames but instead output an asynchronous stream of "events," each corresponding to a brightness change at a specific pixel, timestamped with microsecond-level precision. Fusing this high-speed temporal information with the processing capabilities of an SNN is a primary goal of the field.

P-ETG provides an end-to-end framework for managing uncertainty in these sensor-to-processor pipelines. The timestamps from the DVS are not perfect; they are subject to their own sources of noise and jitter. Furthermore, the clock of the sensor may not be perfectly synchronized with the clock of the neuromorphic processor. P-ETG allows these uncertainties to be modeled as event clouds at the sensor level and then propagated through the SNN. A neuron in the network can be designed to weigh incoming synaptic inputs not only by their synaptic strength but also by the temporal certainty of the originating spike. An event with high timing uncertainty (a wide probability distribution) could be given less influence than a precisely timed event. This enables the design of systems that are robust to sensor noise and clock drift, making decisions based on the most reliable and timely information available.

This application of P-ETG creates a unified framework for both the analysis and synthesis of neuromorphic systems. Analytically, it can be used to verify the robustness of an existing SNN design against the known noise characteristics of a target hardware platform. Synthetically, it can inspire the design of new, uncertainty-aware learning rules and network architectures. For example, a spike-timing-dependent plasticity (STDP) rule could be modified to modulate synaptic weight changes based not just on the mean time difference between pre- and post-synaptic spikes, but also on the consistency and variance of that timing. A synapse would be maximally strengthened only if the pre-synaptic neuron reliably and with low jitter fires just before the post-synaptic one. This approach directly connects the low-level physical properties of the hardware, as modeled by P-ETG, to the high-level learning algorithms that shape the network's function. This suggests a new co-design paradigm where trade-offs between hardware precision (which impacts power and area) and algorithmic sophistication can be formally and quantitatively evaluated, leading to more efficient and powerful neuromorphic systems.

Practical Applications and Benefits in Distributed Systems

Distributed systems, by their very nature, are arenas of concurrency and uncertainty. Composed of multiple autonomous nodes communicating over unreliable networks with unsynchronized clocks, they present fundamental challenges to achieving consistency, fairness, and high performance. Traditional approaches often attempt to tame this complexity by imposing strong synchronization or adopting conservative, worst-case assumptions. Probabilistic Event-Time Geometry offers a transformative alternative: a theoretical foundation for building systems that embrace, quantify, and manage uncertainty, leading to protocols that are demonstrably fairer, more efficient, and more resilient.

Achieving Fairness Under Uncertainty: Probabilistic Fair Ordering

In many distributed applications, particularly competitive ones like financial exchanges, online advertising auctions, or blockchain transaction processing, the order in which events are processed is critically important. The principle of "fair ordering"—that events should be processed in the chronological order of their generation (first-come, first-served)—is a highly desirable property. However, achieving true fair ordering is fundamentally impossible in a system with unsynchronized clocks, as there is no single, globally agreed-upon timeline to serve as a reference. Traditional systems resort to a centralized sequencer, which orders events based on their arrival time at the sequencer, an order that can be arbitrary and influenced by network topology rather than true generation time.

P-ETG provides the core mechanism to solve this long-standing problem through probabilistic fair ordering. By modeling each client's clock as having a random offset from some ideal reference time, the system can compute the "likely-happened-before" probability, $P(e_1 \to e_2)$, for any pair of events from different clients. This probability represents the system's best estimate of their true chronological order, based on all available timing evidence (local timestamps and clock offset distributions). A protocol can then be designed to order events based on these pairwise probabilities. For instance, a directed graph of events can be constructed where an edge from e_1 to e_2 exists if $P(e_1 \to e_2) > 0.5$. A topological sort of this graph yields a total order that is, in a statistical sense, the fairest possible arrangement. This approach replaces the arbitrary, centralized ordering with a decentralized, verifiable, and statistically robust notion of fairness.

Optimizing Performance with Probabilistic Scheduling

The design of real-time and performance-critical distributed systems often hinges on scheduling—the allocation of resources to tasks over time. Deterministic scheduling approaches typically rely on Worst-Case Execution Times (WCETs) and worst-case network latencies to provide hard guarantees that deadlines will be met. While this approach is necessary for safety-critical systems like avionics, it leads to massive over-provisioning and chronically low resource utilization in most other domains, as worst-case conditions are, by definition, extremely rare.

P-ETG enables a more efficient paradigm of probabilistic scheduling. Instead of using single, pessimistic worst-case values, the duration of each task and the latency of each communication step are modeled as probability distributions. A probabilistic schedulability analysis can then be performed, for example, by running a Monte Carlo simulation of the entire task graph. The output is not a binary "schedulable/unschedulable" decision, but a full probability distribution of the project's completion time. The system designer can then make an informed decision based on risk tolerance. For example, a scheduler could be configured to accept a workload that has a 99.999% probability of meeting all deadlines. This allows the system to operate at a much higher level of utilization than a deterministically scheduled one, while still providing a quantifiable, and often acceptably small, risk of occasional deadline misses.

System Integrity and Causal Fidelity

The causal structure of a distributed system is fundamental to its correct operation; writes must be seen after their corresponding reads, replies must follow requests. We can leverage P-ETG to define a novel, high-level metric for system health: **Causal Fidelity**. This metric can be defined as the system-wide probability that the observed orderings of events respect their true, underlying causal relationships, as determined by the probabilistic geometry.
Causal Fidelity can be monitored in real-time by sampling pairs of potentially related events (e.g., from system logs) and computing their causal influence probability. A healthy system operating under normal conditions should exhibit a high Causal Fidelity score. A sudden drop in this metric could serve as a powerful anomaly detector, indicating a potential systemic problem such as a widespread increase in network jitter, a significant clock desynchronization event across a cluster of nodes, or even a malicious attack designed to manipulate event ordering. This provides a principled, physics-based method for monitoring system integrity that goes beyond simple thresholding on latency or error rates, as it directly measures the coherence of the system's causal fabric.

Designing Resilient Protocols

By explicitly modeling sources of uncertainty, P-ETG allows for the design of distributed protocols that are resilient and adaptive by construction. Instead of relying on fixed, conservative parameters, protocols can be designed to respond to the measured statistical properties of the environment. For example, a timeout mechanism in a consensus algorithm is typically set to a large, fixed value to accommodate worst-case network latency. This can slow down the protocol in the common case where latency is low. A P-ETG-informed protocol could dynamically set its timeouts based on the current, continuously measured probability distribution of message round-trip times. It might set the timeout to the 99.9th percentile of the observed latency distribution, allowing it to run faster under good network conditions and automatically become more patient during periods of high congestion, all while maintaining a predictable, quantifiable probability of erroneous timeouts.
This overall approach represents a fundamental philosophical shift in distributed systems design, moving from a paradigm of *preventing* uncertainty to one of *managing* it. Traditional methods focus immense effort on achieving the tightest possible clock synchronization in an attempt to approximate a deterministic, global-time system—a goal that is both expensive and, by the laws of physics and the CAP theorem, ultimately unattainable. P-ETG-based designs, in contrast, accept that uncertainty is an irreducible feature of the distributed world and provide the mathematical tools to reason and make statistically optimal decisions in its presence. This has

profound implications for trust and auditability. In a financial or legal dispute over the order of two transactions, one could use P-ETG to compute the objective, verifiable probability that one transaction truly preceded the other, based on all available timing evidence. This transforms the concept of "fairness" from a vague operational promise into a mathematically rigorous and auditable property of the system.

Conclusion

This analysis has charted the evolution of Event-Time Geometry from its deterministic origins, rooted in an analogy to relativistic spacetime, to a comprehensive probabilistic framework capable of modeling the inherent uncertainty of real-world computational systems. The journey from Deterministic ETG (D-ETG) to Probabilistic ETG (P-ETG) is marked by a systematic transformation: precise event coordinates become probability distributions or "event clouds"; sharp causal boundaries dissolve into fuzzy, probabilistic gradients; and absolute causal relations are replaced by graded measures of confidence. This extension is not merely an addition of noise to an existing model but a fundamental reconceptualization that yields a richer, more physically grounded, and ultimately more applicable theory of causality in complex systems.

The power of P-ETG is amplified by its deep connections to established mathematical disciplines. It finds its natural language in the formalisms of Information Geometry for measuring the distance between uncertain events, in Stochastic Differential Geometry for modeling the dynamics of accumulating uncertainty like clock drift, and in the theory of Probabilistic Graphical Models for performing complex inference on entire networks of causally related events. This robust theoretical underpinning ensures that P-ETG is not an ad-hoc solution but a principled framework that unifies geometry, probability, and information theory.

When situated within the broader landscape of temporal reasoning, P-ETG distinguishes itself by its expressive power. It moves beyond the purely logical ordering of Lamport clocks by introducing a metric, and it offers a more nuanced view of uncertainty than the conservative, worst-case bounds of systems like TrueTime. By modeling the full distribution of uncertainty, it enables a new class of algorithms and protocols that can make statistically optimal decisions, balancing performance and reliability based on quantifiable risk.

The practical benefits of this framework are most evident in two of the most challenging and forward-looking domains of computer science:

- In **neuromorphic systems**, where computation is encoded in the precise timing of stochastic spike events, P-ETG provides the essential tools to analyze the robustness of temporal codes, infer causal connectivity from noisy data, and design systems that are resilient to the inherent jitter of their physical substrate.
- In **distributed systems**, P-ETG offers a foundation for solving long-standing problems in fairness, performance, and resilience. It enables the design of probabilistically fair ordering protocols that eliminate the arbitrariness of centralized sequencers, facilitates high-performance probabilistic scheduling that avoids the over-provisioning of deterministic methods, and provides novel metrics like Causal Fidelity for monitoring system integrity.

Looking forward, as computational systems become increasingly decentralized, asynchronous, and intertwined with the stochastic physical world, the ability to reason rigorously under temporal and spatial uncertainty will cease to be a niche requirement and will become a central design challenge. Probabilistic geometries like P-ETG represent a necessary theoretical foundation for this next generation of computing. From brain-inspired artificial intelligence that must operate on noisy sensor data to global-scale distributed ledgers that must achieve fair consensus among untrusting parties, the principles of P-ETG provide a coherent and powerful language for designing the robust, intelligent, and trustworthy systems of the future. It extends the elegant structure of spacetime to the domain of stochastic information processing, ensuring that even as we embrace randomness, we retain a consistent and quantifiable geometry of cause and effect.

Works cited

1. Lecture 4: Physical and Logical Time, Causality, https://cseweb.ucsd.edu/classes/sp16/cse291-e/applications/ln/lecture4.html 2. Event Ordering in Distributed System - GeeksforGeeks, https://www.geeksforgeeks.org/computer-networks/event-ordering-in-distributed-system/ 3. Features of Hard Real Time Determinism - IntervalZero, https://www.intervalzero.com/features-of-hard-real-time-determinism/ 4. RETHINKING TIME AND DETERMINISM: WHAT HAPPENS TO DETERMINISM WHEN YOU TAKE RELATIVITY SERIOUSLY - Jenann Ismael, https://www.jenanni.com/wp-content/uploads/Rethinking-Time-and-Determinism.pdf 5. Causal Determinism - Stanford Encyclopedia of Philosophy, https://plato.stanford.edu/entries/determinism-causal/ 6. Truetime | Kevin Sookocheff, https://sookocheff.com/tags/truetime/ 7. Beyond Lamport, Towards Probabilistic Fair Ordering - arXiv, https://arxiv.org/html/2510.13664v1 8. [Literature Review] Fair Ordering - Moonlight, https://www.themoonlight.io/en/review/fair-ordering 9. Probabilistic Schedule Management - Real Options Valuation, https://rovdownloads.com/probabilistic-schedule-management/ 10. On Assessing the Viability of Probabilistic Scheduling with Dependent Tasks - UPCommons, https://upcommons.upc.edu/bitstreams/f591ec06-fd77-4068-9792-5c0c6fa644fd/download 11. Metrics for Probabilistic Geometries - DTU, https://www2.compute.dtu.dk/~sohau/papers/uai2014/gplvm_metricUAI14.pdf 12. Probabilistic Approach to Geometry - Project Euclid, https://projecteuclid.org/proceedings/advanced-studies-in-pure-mathematics/probabilistic-approach-to-geometry/toc/10.2969/aspm/05710000 13. Geometry 13.2a, Theoretical Probability & Complement of an event - YouTube, https://www.youtube.com/watch?v=BeP81E--ajc 14. CS 228 - Probabilistic Graphical Models - Stanford Computer Science, https://cs.stanford.edu/~ermon/cs228/index.html 15. Probabilistic Graphical Models: A Gentle Intro - DZone, https://dzone.com/articles/probablistic-graphical-models-a-gentle-intro 16. Graphical Models for Probabilistic and Causal Reasoning, https://ftp.cs.ucla.edu/pub/stat_ser/r236-3ed.pdf 17. Learning Temporal Causal Graphs for Relational Time-Series Analysis, https://icml.cc/Conferences/2010/papers/235.pdf 18. 29 Probabilistic Graphical Models - Foundations of Computer Vision - MIT, https://visionbook.mit.edu/graphical_models.html 19. T-CPDL: A Temporal Causal Probabilistic Description Logic for Developing Logic-RAG Agent - arXiv, https://arxiv.org/pdf/2506.18559 20. Project Scheduling Techniques: Probabilistic and Deterministic - Advaiya, https://advaiya.com/wp-content/uploads/2025/08/White-Paper-on-Project-Scheduling-Techniques-1.pdf 21. Causal Consistency Model in System Design - GeeksforGeeks, https://www.geeksforgeeks.org/system-design/causal-consistency-model-in-system-design/ 22. POINT PROCESSES - UCLA Statistics & Data Science, http://www.stat.ucla.edu/~frederic/papers/pointprocesses.pdf 23. Temporal Point Processes -

MPI-SWS Courses, https://courses.mpi-sws.org/hcml-ws18/lectures/TPP.pdf 24. Fifty years later: new directions in Hawkes processes - Idescat, https://www.idescat.cat/sort/sort461/46.1.1.Worrall-etal.pdf 25. Hawkes process - Wikipedia, https://en.wikipedia.org/wiki/Hawkes_process 26. Geometric Hawkes Processes with Graph Convolutional ... - Jin Shang, https://jshang2.github.io/pubs/GeometricHawkes.pdf 27. Hawkes Models and Their Applications - Annual Reviews, https://www.annualreviews.org/doi/pdf/10.1146/annurev-statistics-112723-034304 28. Hawkes Processes Modeling, Inference and Control: An Overview - ResearchGate, https://www.researchgate.net/publication/346475299_Hawkes_Processes_Modeling_Inference_and_Control_An_Overview 29. Geometric Hawkes Processes with Graph Convolutional Recurrent Neural Networks, https://ojs.aaai.org/index.php/AAAI/article/view/4416/4294 30. Equality Constraints in Linear Hawkes Processes - Proceedings of Machine Learning Research, https://proceedings.mlr.press/v177/mogensen22a/mogensen22a.pdf 31. [2501.05535] On Fair Ordering and Differential Privacy - arXiv, https://arxiv.org/abs/2501.05535 32. Neuromorphic Event-based Sensing and Computing - PeAR WPI, https://pear.wpi.edu/eventvision.html 33. integrating event-based neuromorphic processing and hyperdimensional computing with tropical algebra - IJCI Conference Proceedings, https://ijcionline.com/paper/13/13624ijci03.pdf 34. Neuromorphic computation using event-based sensors : from algorithms to hardware implementations | Request PDF - ResearchGate, https://www.researchgate.net/publication/337936764_Neuromorphic_computation_using_event-based_sensors_from_algorithms_to_hardware_implementations 35. An Application-Driven Survey on Event-Based Neuromorphic Computer Vision - MDPI, https://www.mdpi.com/2078-2489/15/8/472 36. 4.1 Fairness - Process Analysis Toolkit, https://pat.comp.nus.edu.sg/resources/OnlineHelp/htm/scr/4%20Special%20Features/4.1%20Fairness.htm 37. Securing causal relationships in distributed systems - Michael Reiter, https://reitermk.github.io/papers/1995/CJ.pdf 38. Causal consistency - Wikipedia, https://en.wikipedia.org/wiki/Causal_consistency 39. Distributed Systems 3.3: Causality and happens-before - YouTube, https://www.youtube.com/watch?v=OKHIdpOAxto