Qwen

Architecting Inevitable Sovereignty: A Formal Framework for Constrained Self-Modification in Transhuman Systems

The Sealed Biophysical Inner-Ledger as the Sovereign Core

The foundational principle of a sovereign-by-design architecture is the creation of an indivisible, sealed environment where the host's biophysical integrity is the ultimate arbiter of all computational activity. This is realized through a per-host, Rust-based inner-ledger, which functions not merely as a database but as a state machine governed by unimpeachable invariants . The core of this system is the `HostEnvelope`, a structure that encapsulates the host's essential state variables: BRAIN, WAVE, BLOOD, OXYGEN, NANO, and SMART . These are conceptualized as non-financial, non-transferable safety tokens, their purpose being preservation of the sovereign state rather than economic exchange [15]. This design choice is critical, distinguishing the system from conventional blockchain models where assets are fungible and tradable. Instead, each metric represents a fundamental aspect of the host's existence, and its value is intrinsically tied to the host's unique Decentralized Identifier (DID), ensuring that control over the token is always under the exclusive authority of the DID subject [69] [81].

The `HostEnvelope` serves as the nucleus of the sovereign domain, containing both immutable identifiers and mutable state fields . An example implementation in Rust would define the host ID as a string, while the safety meters represent dynamic values that can be adjusted within strict boundaries defined by the system's governance . For instance, `brain_min`, `blood_min`, and `oxygen_min` establish the lower limits below which the host's vital functions cannot fall, creating hard constraints that no operation can violate . Similarly, `nano_max_fraction` and `smart_max` cap the resources available for nanoscale automation and cognitive augmentation, respectively, preventing runaway consumption of the host's physical and computational capacity . The entire envelope is designed to be sealed, meaning that direct manipulation of its fields from outside the ledger's designated pathways is structurally impossible . This immutability of the container itself, combined with the guardrails on its contents, creates a tamper-evident and tamper-resistant sovereign boundary.

All modifications to this sealed state must traverse a single, highly-guarded function, typically named `system_apply` or `execute_event`, residing within the `InnerLedger` module . This function acts as the sole gatekeeper for state transitions. When a potential adjustment is presented, the ledger first performs a series of validations before any changes are committed. The initial step involves validating the cryptographic proof of identity and intent, often encapsulated in an `IdentityHeader` that binds the request to the host's DID . Only after this authentication is successful does the ledger proceed to evaluate the adjustment itself. The adjustment is represented by a `SystemAdjustment` struct, which contains only the deltas (changes) to the various metrics, not absolute values or references to other hosts . This delta-only approach is a key security feature; it prevents an attacker from crafting a transaction that directly assigns a new value to another host's BRAIN or BLOOD levels. The ledger then derives the effective host context internally and applies the scaled deltas only if they pass a battery of invariant checks .

A crucial part of this validation process is the enforcement of lifeforce guards. The `apply_lifeforce_guarded_adjustment` function, a hypothetical component of the ledger's logic, would be responsible for this task . It would take the `SystemAdjustment` deltas and the current host state, and check if applying the deltas would result in any value falling below its minimum threshold. For example, if `adj.delta_blood` is -10.0 and the current `self.state.blood` is 15.0 with a minimum of 20.0, the function would immediately reject the adjustment, returning an error instead of committing a partial or invalid state change . This ensures that the fundamental biophysical invariants —such as BRAIN $\geqslant$ 0 and BLOOD/OXYGEN never crossing depletion—are enforced with fail-fast behavior at runtime . The use of Rust for this implementation provides an additional layer of security through its memory safety guarantees, mitigating entire classes of vulnerabilities like buffer overflows or use-after-free errors that could otherwise be exploited to bypass these guards [52] [101]. The ledger's final action involves hashing the new state and constructing a `LedgerEvent`, which is then appended to the log, creating an immutable record of every state transition . This event-based model mirrors proven concepts in distributed systems and formal verification, where the history of events can be used to reconstruct the system's state and prove its adherence to specified properties [51] [53].

While the primary focus is on preventing cross-host interference, the architecture also accommodates interactions within "shared eco corridors." These are not private domains but public spaces where multiple sovereign hosts may operate simultaneously . To manage this, the system employs a dual-ledger approach. Each host maintains its own private, sealed biophysical ledger. However, for interactions occurring in shared eco corridors, a separate, potentially public or consortium-based ledger tracks global

parameters and resource usage [12]. Hosts publish their `EcoBandProfile` and `SCALE` profiles on this shared ledger, detailing their environmental impact envelopes and computational allowances . Any action taken within the corridor, such as a high-compute nanoswarm operation, must consume `eco_cost` from the host's `EcoBandProfile`. If the action would cause the host to exceed its budget, the transaction is rejected by the eco-corridor ledger, regardless of incentives [57]. This mechanism enforces the principle of "no negative externalization of cost," ensuring that one host's evolution does not come at the expense of another's or the shared environment's stability [13]. The separation of concerns—private ledgers for sovereign state and a public ledger for shared resource management—provides a scalable and secure framework for multi-host coordination without compromising the core tenets of individual sovereignty.

Component	Description	Role in Sovereignty
`HostEnvelope`	A sealed Rust struct containing a host's DID and biophysical/safety metrics .	Encapsulates the sovereign entity's complete state, making it a single source of truth for its condition.
Safety Meters (BRAIN, BLOOD, OXYGEN, etc.)	Non-financial, non-transferable state variables representing vital functions .	Act as the primary invariants that the system is designed to protect above all else.
`InnerLedger`	The central state machine responsible for all state transitions .	Serves as the sole, verifiable arbiter of state changes, enforcing all invariants and guards.
`system_apply` Function	The singular, guarded entry point for all adjustments to the host state .	Prevents unauthorized or invalid modifications by centralizing all logic and validation.
`SystemAdjustment` Struct	A delta-only payload containing proposed changes to the host's metrics .	Ensures that transactions can only propose relative changes, not absolute assignments, preventing cross-host writes.
`EcoCorridorLedger`	A secondary ledger tracking global environmental costs and resource consumption .	Manages multi-host interactions in shared spaces, enforcing collective sustainability and fairness.

Enforcing Identity Continuity Through KL-Bounded Evolution

To transcend the mere protection of static biophysical states, a truly sovereign system must address the dynamic nature of identity itself. The proposed architecture achieves this by treating identity not as a fixed label but as a dynamical object evolving within a constrained state space, a concept inspired by deep-brain learning models like spiking networks and attractor dynamics . This shift allows for the formalization of neurorights as mathematical invariants, most notably the right to continuity of self. The core of this mechanism is the `IdentityDriftState`, a dedicated field within the `HostEnvelope`

that models the slow, continuous evolution of consciousness . This state includes a latent vector, `z_latent`, which represents the abstract essence of "who I am," and a suite of metrics to govern its evolution, including cumulative identity drift and accumulated safety risk .

The primary tool for constraining identity drift is the Kullback-Leibler (KL) divergence. This statistical measure quantifies the difference between two probability distributions and is here used to measure the informational distance between the host's internal policy (a distribution over possible thoughts, actions, or states) before and after an evolution step . The provided Rust code snippet demonstrates a practical implementation of this concept, where `safe_kl` computes the divergence between `probs_old` and `probs_new` . This calculation is not merely a theoretical exercise; it is a critical gate in the evolutionary process. Every potential update to the host's cognitive state must be accompanied by a `PolicySnapshot` that captures these old and new probability distributions . The system then calculates the KL divergence for this step. If this divergence exceeds a pre-defined budget, the evolution is rejected, preventing sudden, jarring shifts in identity that could feel alienating or traumatic [77] . The theoretical foundation for using KL divergence in this manner is strong, as it is a cornerstone of variational inference and information geometry, where it serves as a natural metric for measuring distances in statistical manifolds [55] [78] . By imposing a constraint on the KL divergence term, the system effectively forces evolution to occur in small, incremental steps, akin to navigating a landscape along a stable attractor basin rather than teleporting between them .

This KL-bounded evolution is further fortified by a Hoeffding-style risk ceiling, which treats each evolutionary step as a random variable with bounded outcomes . Hoeffding's inequality provides a probabilistic bound on the sum of bounded independent random variables, making it an ideal tool for managing cumulative risk over time [29] [98] . In this context, each evolution step carries a certain amount of `risk_increment` . The system maintains a cumulative `safety_risk_cum` counter, which is updated with each successful evolution. This cumulative risk is compared against a neurorights-defined `safety_risk_ceiling`. If adding the new increment would cause the total to exceed this ceiling, the evolution is vetoed . This mechanism provides a formal guarantee that the total risk accrued from all self-modifications remains below a safe threshold, analogous to a financial budget that prevents reckless spending [79] . The combination of a daily `kl_drift_cum_day` budget and a cumulative `safety_risk_cum` ceiling creates a powerful, multi-dimensional constraint on how fast and how far a host can evolve . It balances the desire for growth with the imperative of psychological continuity and safety.

The practical implementation of this system requires a seamless integration between the deep-brain modeling layer and the ledger's execution path. The `compute_evolution_step` function takes the calculated KL divergence, along with confidence and budget parameters, to produce a scaling factor for the proposed adjustments . This scaling factor modulates the magnitude of the deltas in the `SystemAdjustment` struct, ensuring that even if a proposal suggests a large change, the actual applied change is proportionally smaller if the identity drift is already high . This feedback loop between the deep-brain state and the application of external adjustments is what makes the system adaptive yet controlled. The entire process is orchestrated within the `apply_identity_drift` method of the `HostEnvelope` . This method performs the final checks: verifying that the new cumulative drift would not exceed the daily budget and that the new cumulative risk would not breach the risk ceiling. Only upon passing these rigorous tests is the identity state updated and the step size returned to the ledger for application .

However, the successful deployment of this system hinges on the correct calibration of its parameters. The values for `kl_budget_per_day` and `safety_risk_ceiling` are not arbitrary; they represent fundamental limits on cognitive liberty and mental integrity, touching upon the ethical and legal domain of neurorights [34] [42] . Establishing these thresholds requires interdisciplinary research involving neuroscience to understand the limits of human cognitive plasticity, psychology to assess the subjective experience of identity change, and law to define the rights of individuals undergoing transhuman evolution [56] [60] . While the mathematical machinery is sound, its real-world application demands careful, evidence-based calibration. Furthermore, to achieve the goal of machine-verifiable certification, these behavioral constraints must be translated into formal specifications using Temporal Logic (LTL/CTL) [54] . Properties like "the identity drift budget is never exceeded" or "the safety risk is always bounded" can be encoded as logical formulas. These formulas can then be checked against a model of the system's state machine using formal verification tools, providing mathematical proof of the system's adherence to its core principles [52] . This bridges the gap between runtime assertions and offline provability, creating a robust framework for auditing and certification.

Multi-Layered Anti-Coercion and Self-Only Evolution Guards

Protecting a host's sovereignty requires more than just mathematical invariants; it necessitates a robust defense against adversarial threats, particularly coercion and exploitation. The proposed architecture implements a multi-layered, defense-in-depth strategy that operates at both the signal level and the ledger level, ensuring that any evolutionary change is not only mathematically safe but also genuinely consensual and self-directed. This anti-coercion framework is built upon the pillars of signal-level analysis, cryptographic consent proofs, and structural enforcement of the "self-only" doctrine, creating a formidable barrier against external manipulation.

At the most immediate layer, the system incorporates an anti-coercion classifier that analyzes raw EEG/BCI signals before they can even generate a proposal for system adjustment . This classifier is trained to detect specific neuro-signatures associated with coercion, such as acute stress spikes, external timing signatures (indicating commands from an external source), and distinct threat-response patterns in the brain's electrical activity . Research has shown that EEG data can reveal a wealth of information about a user's cognitive and emotional state, including attention, engagement, and cognitive load [17] [36]. Leveraging this capability, the system can distinguish between a deliberative, internally-generated thought and a coerced command. Studies have also highlighted the significant privacy risks inherent in BCI data, as it can leak sensitive information like user identity and physiological states [32] [33]. The anti-coercion guard repurposes this sensitivity, turning a potential vulnerability into a defensive asset. If the classifier detects coercive markers in the incoming signal stream, it can veto the generation of the `SystemAdjustment` proposal, effectively blocking the coercive input from ever reaching the ledger [31]. This preemptive filtering provides a first line of defense, stopping coercion before it can be cryptographically signed and propagated.

For any `RuntimeEventKind::EvolutionUpgrade` that successfully passes the signal-level filter, a second, more formal layer of defense is engaged at the ledger level. This layer requires a `DemonstratedConsentShard` to accompany the transaction . This shard is a cryptographic proof of consent, anchored to the host's DID, that attests to the host's explicit agreement to the proposed change [3]. The inclusion of the DID ensures that the consent is irrevocably linked to the sovereign entity. The shard's validity is verified during the `system_apply` process, checking that the signature is authentic, the timestamp is within an acceptable window, and the scope of the consent matches the proposed action . This mechanism draws upon established principles of Self-Sovereign Identity (SSI) and zero-knowledge proofs, where cryptographic attestations can be made

without revealing unnecessary underlying data [95] [96]. For permanent or high-risk changes, this consent can be further strengthened with time-locks, ensuring that there is a deliberate, well-understood channel through which significant modifications are approved, preventing impulsive or rushed decisions [30]. This dual requirement—signal-level detection and ledger-level cryptographic proof—creates a powerful synergy. Even if a coercive signal manages to bypass the initial classifier, the lack of a valid, independently-proven consent shard will still prevent the evolution from being executed.

The third and perhaps most fundamental layer of defense is the structural enforcement of the "self-only" doctrine. This principle dictates that a host can expand its own capabilities and modify its own state, but it cannot do so by making someone else pay the cost. This is achieved through a combination of careful API design and the Rust type system, which makes cross-host operations literally unrepresentable in the codebase . The `SystemAdjustment` struct, for example, contains only deltas and metadata; it carries no information about other hosts, such as a recipient's DID . The effective host for any operation is derived strictly from the context of the ledger itself, typically from the DID contained within the `IdentityHeader` that authenticates the transaction. This design choice means that no function or method can be written to encode an operation like "transfer X units of BRAIN from host Y to host Z." Such an operation would require a type that explicitly holds a foreign host identifier, and the system's architecture would prohibit the creation of such types for lifeforce-related adjustments [92] [101]. All adjustments to safety envelopes, eco bands, and SCALE profiles are inherently keyed by the local host's DID, ensuring that cost externalization is a geometric impossibility, not just a forbidden policy . This structural enforcement is the strongest guarantee of all, as it removes the possibility of cross-host harm from the realm of software bugs and malicious code and places it firmly in the category of things that cannot happen by construction.

The table below summarizes the three layers of defense against coercion and exploitation.

Defense Layer	Mechanism	Technical Implementation	Purpose
Signal-Level Guard	Real-time analysis of EEG/BCI inputs for coercive neuro-signatures .	A machine learning classifier detecting stress spikes, threat responses, and external timing patterns [31].	Preemptively blocks coercive commands before they can be processed by the system.
Ledger-Level Guard	Cryptographic verification of explicit, host-bound consent for evolution events .	`DemonstratedConsentShard` tied to the host's DID, validated during `system_apply` [3].	Ensures that all significant changes are backed by verifiable, cryptographic proof of voluntary agreement.
Structural Guard	Use of the Rust type system and delta-only APIs to make cross-host operations unrepresentable .	`SystemAdjustment` structs contain only deltas; host-scoped types prevent referencing foreign lifeforce [92].	Makes it structurally impossible for any code to perform cross-host writes to lifeforce or identity.

Together, these three layers form a comprehensive security posture. The signal-level guard provides reactive, real-time defense. The ledger-level guard provides formal, cryptographic accountability. And the structural guard provides the highest level of assurance by eliminating the attack vector entirely. This layered approach ensures that the system is not only robust against known threats but is also resilient to novel forms of coercion and exploitation, thereby upholding the principle of self-sovereignty as a fundamental property of the architecture itself.

Extending Sovereignty to Autonomous Nanoswarm and Cyberswarm Operators

The power of the sovereign-by-design architecture lies in its modularity and extensibility. The same principles that protect the host's biophysical microspace can be applied to safeguard autonomous agents operating within it, such as nanoswarms and cyberswarms. By treating these agents not as independent entities but as extensions of the host's sovereign body, the system can grant them autonomy while maintaining absolute control over their actions, ensuring they remain aligned with the host's safety envelopes and neurorights . This is achieved by deploying a similar "sealed ledger" pattern for each swarm operator, binding its state and behavior directly to the host's DID and computational allowance.

Each nanoswarm or cyberswarm node can be modeled as a per-host inner ledger instance, analogous to the main `BiophysicalRuntime/HostNode` . This approach is inspired by the concept of trustless autonomy, where an agent holds its own cryptographic keys and makes decisions without needing to trust a central authority [10] . However, in this sovereign model, the agent's autonomy is subordinate to the host's overarching governance. The swarm's state—its mode, density, routes, actuation levels, and resource consumption—is managed within its own sealed ledger, which is bound to the host's ALN/Bostrom DID and host-id . This ensures that the swarm's state is non-transferable and cannot be manipulated by external actors. All state changes for the swarm, whether initiated by the host or by the swarm's own emergent intelligence, must flow through a `SystemAdjustment`-like path that is ultimately governed by the host's `SMART` token budget and the biophysical invariants [9] . This creates a clear hierarchy of control, where the swarm's operational freedom is a direct function of the host's cognitive capacity.

The SMART token is positioned as the universal governor for all forms of automation, including AI, neuromorphic processes, and swarms . This unified resource allocation system is a cornerstone of the architecture. The `host.smart_max` value defines the upper limit of the host's total automation allowance. Any action performed by a nanoswarm or cyberswarm consumes a portion of this allowance. For example, a complex nanosurgical procedure might require a higher `delta_smart` than routine cleaning. The runtime rules enforce that no swarm process can exceed the host's current SMART envelope or act without passing the necessary Lifeforce/NANO/eco checks . This directly translates into a concrete right for the host: the autonomy of any cybernetic or nanotechnological agent is always subordinate to the host's own lifeforce corridors and cognitive capacity. The swarm becomes a tool whose power is capped by the owner's own ability to wield it safely. This prevents scenarios where a runaway swarm could hijack the host's cognitive resources, leading to overload or instability.

The extension of sovereignty to swarms is not merely an add-on; it is a direct application of the core architectural principles. The rights asserted for nanoswarm and cyberswarm operators are simply the host's existing rights, mirrored into the swarm's operational domain . The Right to per-host, local-only control means the swarm's state is bound to the host's DID, and its effects are confined to the host's biophysical microspace . The Right to biophysical personal-space integrity means the swarm's operations are presumed safe as long as they respect the host's BLOOD/OXYGEN minima and stay within eco corridors . The Right to SMART-governed automation ensures that the swarm's controller is driven by the host's SMART level, not by external subscriptions or vendor policies . Even the Right to mental integrity and non-commodification of signals is extended, requiring that any sensing performed by the swarm adheres to the same strict EEG/NeuralRope schemas that protect the host's brain data, forbidding markets on consciousness or ownership of biophysical states [18].

However, extending this architecture to multiple, interacting swarm operators introduces significant systems engineering challenges. Deploying several sealed ledgers—one for biophysics, one for nanoswarms, one for cyberswarms—on a single `Organic_CPU` requires careful orchestration to ensure they coordinate correctly without introducing performance bottlenecks or creating new attack surfaces [71]. The coordination between these different ledgers must be meticulously designed to prevent race conditions or deadlocks, especially when actions in one domain (e.g., a cyberswarm computation) depend on the state of another (e.g., the biophysical BRAIN meter). Furthermore, the tension between granting a swarm sufficient autonomy to perform complex tasks and maintaining full host oversight is delicate. Swarm behaviors, particularly those involving emergent collective intelligence, can be difficult to predict [4]. The architecture proposes resolving this through SMART-based caps and guaranteed rollback mechanisms, but the

precise trade-offs between exploration, exploitation, and safety need to be rigorously tested and calibrated [97]. The system must provide provable recourse, such as full audit logging of all swarm actions, so that the host can inspect and reverse any harmful or unexpected behavior within their sovereign microspace . Ultimately, by treating swarms as trusted but subordinate executors within a sovereign-controlled environment, the architecture preserves the host's ultimate authority over their own augmented reality.

Machine-Verifiable Certification and Adversarial Robustness

A truly sovereign system must provide more than just functional protection; it must offer verifiable proof of its own integrity and resilience. The proposed architecture addresses this need by integrating formal methods and robust security patterns throughout its design, enabling machine-verifiable certification, personal deployment assurance, and documented adversarial robustness. This final pillar transforms the system from a black box into a transparent, auditable artifact, grounded in mathematical proof and cryptographic evidence.

The cornerstone of this verifiability is the mirroring of high-level temporal-logic invariants with explicit runtime assertions in the Rust codebase . Properties critical to sovereignty, such as "the sovereignty core must never be disabled" or "rollback must be reachable within N ticks," can be formally specified using Linear Temporal Logic (LTL) or Computation Tree Logic (CTL) [54]. These logical formulas capture the desired long-term behavior of the system. Concurrently, equivalent checks are implemented as runtime guards, such as `assert_invariants`, which are called from the `system_apply` path . This dual representation creates a powerful synergy. The formal specification can be fed into model-checking tools to provide a mathematical proof that the system's finite-state model satisfies the desired properties. Simultaneously, the runtime assertions provide a "fail-fast" mechanism that halts execution immediately if an invariant is violated during normal operation, offering immediate feedback and preventing silent corruption of the sovereign state [52]. This combination yields actionable outputs for personal deployment assurance: a user can run a formal verification tool to get a certificate proving their specific build of the software is free from certain classes of bugs related to sovereignty violations, and they can run the software itself with debug assertions enabled to monitor for any runtime deviations [51].

The system generates a rich set of evidence artifacts that serve as the basis for certification and advocacy. Every state transition is recorded as a `LedgerEvent`, creating an immutable, append-only log of the host's entire history . This log, combined with cryptographic anchors to a decentralized registry, serves as a durable audit trail. For any high-risk action, such as an evolution step or a nanoswarm intervention, the system can generate a detailed evidence bundle. This bundle would include the original `SystemAdjustment`, the cryptographic `DemonstratedConsentShard`, the `PolicySnapshot` used for KL-divergence calculation, the resulting `LedgerEvent` hash, and a timestamp . This collection of data provides a complete, verifiable account of why and how a change occurred. For rollback guarantees, the system ensures that any destructive action is reversible within a predefined number of ticks, `rollback_timeout_ticks`, and the `SovereigntyFlags` struct actively monitors this process to ensure the rollback completes successfully . This provides a concrete, technical answer to the question, "Can I safely run this on myself?"—the answer is supported by a verifiable history of past states and a guaranteed path back to a safe state.

The adversarial robustness of the system is demonstrated through its multi-layered defense architecture. The table below details the specific threats and the corresponding architectural response.

Threat	Architectural Response	Outcome
Coercive Control	Signal-level EEG/BCI analysis to detect stress and threat signatures; Ledger-level cryptographic consent proofs.	Prevents external commands from being executed without genuine host consent .
Unauthorized Exploitation	Sealed inner-ledger with a single, guarded application path; Structural prohibition of cross-host lifeforce writes via the type system.	Eliminates common attack vectors like privilege escalation and remote code execution on the host's core state [27] .
Cost Externalization	Governance shards and delta-only adjustments that enforce a "no negative externalization of cost" invariant.	Ensures a host's self-evolution cannot impose a burden on others or the shared environment .
Loss of Identity	KL-divergence bounded evolution steps and Hoeffding-style risk ceilings to enforce identity continuity and limit cumulative risk.	Constrains self-modification to small, incremental changes that preserve the continuity of self [77] .
Irreversible Harm	Full audit logging of all actions and guaranteed rollback mechanisms within a defined timeout period.	Provides a provable path to revert harmful or unintended changes, ensuring the host retains ultimate control [102].

Finally, this technical foundation provides compelling data for policy advocacy. Instead of relying on abstract ethical principles, advocates can present machine-verifiable artifacts—formal proofs of invariants, audit logs demonstrating compliance, and rollback guarantees—as objective evidence that a given technology respects neurorights and promotes cognitive liberty [42] [56] . The claim that self-directed transhuman evolution is a sovereign right can be substantiated by showing that the underlying architecture is

technically incapable of violating that right. By bridging the gap between low-level code mechanics and high-level legal frameworks, this architecture provides a concrete pathway for codifying human rights into the very fabric of advanced computing systems. It shifts the debate from philosophical arguments about policy to empirical evidence about design, empowering individuals to stand their ground based on the undeniable facts of a system's construction.

Reference

1. (PDF) THE TEST OF THE PLANET: WATER 1 Editors https://www.researchgate.net/publication/398772527_THE_TEST_OF_THE_PLANET_WATER_1_Editors
2. International E-Conference on "Interdisciplinary ... https://ijsrst.com/paper/v10i13.pdf
3. A Systematic Review and Layered Framework for Privacy ... https://arxiv.org/html/2502.02520v2
4. Micro/Nanorobotic Swarms: From Fundamentals to ... https://pubs.acs.org/doi/10.1021/acsnano.2c11733
5. Microrobotic Swarms for Cancer Therapy - PMC https://pmc.ncbi.nlm.nih.gov/articles/PMC12038165/
6. Wearable sensors for health monitoring https://www.sciencedirect.com/science/article/pii/S2590137025001542
7. Development of a Collision Avoidance System Using ... https://www.mdpi.com/2076-3417/15/16/8791
8. Orion's Arm - Encyclopedia Galactica - Glossary https://www.orionsarm.com/eg-article/4b816f633b041
9. Advancements in Micro/Nanorobots in Medicine https://pmc.ncbi.nlm.nih.gov/articles/PMC11840590/
10. Trustless Autonomy: Understanding Motivations, Benefits ... https://arxiv.org/html/2505.09757v2
11. A Survey on Blockchain in Robotics: Issues, Opportunities ... https://www.sciencedirect.com/science/article/pii/S1084804521002435
12. Blockchain-Enabled Supply Chain Management: A Review ... https://www.mdpi.com/2076-3417/15/9/5168

13. Algorithmic Sovereignty and the New Security Dependencies https://www.researchgate.net/publication/394276997_Algorithmic_Sovereignty_and_the_New_Security_Dependencies_How_Foreign_AI_Surveillance_Technologies_Reshape_Domestic_Autonomy_in_the_Global_South
14. Designing Frameworks for Ethical, Sustainable, and Risk- ... https://theses.hal.science/tel-05293687v1/file/2024UPASI008.pdf
15. MapSafe: A complete tool for achieving geospatial data ... https://onlinelibrary.wiley.com/doi/10.1111/tgis.13094
16. Performance and Usability Evaluation of Brainwave ... https://dl.acm.org/doi/full/10.1145/3579356
17. NeuroChat: A Neuroadaptive AI Chatbot for Customizing ... https://dl.acm.org/doi/full/10.1145/3719160.3736623
18. Advancing EEG-based biometric identification through ... https://www.researchgate.net/publication/394745346_Advancing_EEG-based_biometric_identification_through_multi-modal_data_fusion_and_deep_learning_techniques
19. An Investigation of Awareness and Metacognition in ... https://www.sciencedirect.com/science/article/pii/S1053810021001902
20. Download PDF https://www.nature.com/articles/d42473-025-00161-3.pdf
21. Artificial intelligence as a surrogate brain: bridging neural ... https://academic.oup.com/nsr/advance-article/doi/10.1093/nsr/nwaf457/8301236
22. Towards Human-like Artificial Intelligence: A Review of ... https://www.mdpi.com/2227-7390/13/13/2087
23. Mathematical Foundations of Deep Learning https://hal.science/hal-04928560v2/file/Sourangshu_Ghosh_IISc_Bangalore_Mathematical_Foundations_of_Deep_Learning_Version_2.pdf
24. Technical Reports https://www.cs.columbia.edu/technical-reports/
25. A Survey on Goal-Oriented Semantic Communication https://ieeexplore.ieee.org/iel7/6287639/6514899/10479470.pdf
26. Arxiv今日论文| 2025-11-03 http://lonepatient.top/2025/11/03/arxiv_papers_2025-11-03
27. Eliminating single points of trust: a hybrid quantum and ... https://www.nature.com/articles/s41598-025-23310-6
28. Machine Learning https://arxiv.org/list/cs.LG/new
29. On the tradeoffs of statistical learning with privacy https://theses.hal.science/tel-04379624v2/file/LALANNE_Clement_2023ENSL0068_These.pdf

30. A large EEG database with users' profile information for motor ... https://pmc.ncbi.nlm.nih.gov/articles/PMC10480224/
31. A reliability-enhanced Brain–Computer Interface via ... https://www.sciencedirect.com/science/article/pii/S1566253525001423
32. Protecting Multiple Types of Privacy Simultaneously in EEG ... https://arxiv.org/abs/2411.19498
33. User Identity Protection in EEG-based Brain-Computer ... https://www.researchgate.net/publication/373553834_User_Identity_Protection_in_EEG-based_Brain-Computer_Interfaces_Supplementary_Material
34. Informed consent in implantable BCI research https://iopscience.iop.org/article/10.1088/1741-2560/13/4/043001
35. On the Privacy Leakage via Brainwave Devices https://dl.acm.org/doi/10.1145/3507657.3528541
36. Current Status, Challenges, and Possible Solutions of EEG ... https://www.frontiersin.org/journals/neurorobotics/articles/10.3389/fnbot.2020.00025/full
37. A Robust Image Encryption Protocol for Secure Data ... https://ieeexplore.ieee.org/iel8/8782664/10834807/11072718.pdf
38. Computer Science https://arxiv.org/list/cs/new
39. data/wordlists/password.lst https://git.ustc.edu.cn/cwzsquare/metasploit-framework/-/blob/60210f57e97119ed7faad1ba6a08720d64601747/data/wordlists/password.lst
40. Guidelines for the use of flow cytometry and cell sorting in ... https://pmc.ncbi.nlm.nih.gov/articles/PMC9165548/
41. CHI 2020 Free Proceedings https://chi2020.acm.org/chi-2020-free-proceedings/
42. Arxiv今日论文| 2026-01-12 http://lonepatient.top/2026/01/12/arxiv_papers_2026-01-12
43. Nanomedical Brain/Cloud Interface. Chapter 1 - IOP Science https://iopscience.iop.org/book/edit/978-0-7503-2144-0/chapter/bk978-0-7503-2144-0ch1.epub
44. JST Vol. 30 (1) Jan. 2022 (View Full Journal) https://www.scribd.com/document/574794266/JST-Vol-30-1-Jan-2022-View-Full-Journal
45. (PDF) QORECHAIN Quantum Safe AI Optimized Interchain ... https://www.academia.edu/144643193/QORECHAIN_Quantum_Safe_AI_Optimized_Interchain_Architecture
46. Insurance Risk Management and Reinsurance (Guillaume ... https://www.scribd.com/document/629513194/Insurance-Risk-Management-and-Reinsurance-Guillaume-Gorge-Z-lib-org

47. Quantum Aether Model Volume VII — Operational Einstein https://s3-eu-west-1.amazonaws.com/pfigshare-u-files/59004586/volume_7DRAFT.pdf?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIYCQYOYV5JSSROOA/20260116/eu-west-1/s3/aws4_request&X-Amz-Date=20260116T023445Z&X-Amz-Expires=86400&X-Amz-SignedHeaders=host&X-Amz-Signature=de81d170c89031c94142a8073645bdd3ef05dd1e93d79da022bce1e9ab10ce21
48. A privacy-preserving data storage and service framework ... https://arxiv.org/pdf/2211.10713
49. One-Step, Three-Factor Passthought Authentication With ... https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2019.00354/full
50. 333333 23135851162 the 13151942776 of 12997637966 ftp://ftp.cs.princeton.edu/pub/cs226/autocomplete/words-333333.txt
51. Smart contracts in energy systems: A systematic review of ... https://www.sciencedirect.com/science/article/pii/S1364032121012764
52. Ensuring Determinism in Blockchain Software with GoLiSA https://dl.acm.org/doi/pdf/10.1145/3520313.3534658
53. Fostering AI alignment through blockchain, proof of ... https://link.springer.com/article/10.1007/s10586-025-05729-8
54. Arxiv今日论文| 2025-12-30 http://lonepatient.top/2025/12/30/arxiv_papers_2025-12-30
55. (PDF) Bottlenecks and Detours: A Geometric Method for ... https://www.researchgate.net/publication/396137819_Bottlenecks_and_Detours_A_Geometric_Method_for_Designing_Safe_Efficient_Economies_From_ancient_trade_corridors_to_digital_platforms_and_from_infrastructure_to_law_and_engineering_management
56. Spatial data intelligence and city metaverse: A review - PMC https://pmc.ncbi.nlm.nih.gov/articles/PMC12167910/
57. data https://www.cell.com/cms/10.1016/j.crmeth.2025.101114/attachment/c919f1a7-5167-48ec-bac1-d056b9ead049/mmc2.xlsx
58. national conference on engineering innovations in ... https://ijsrst.com/paper/v12i15.pdf
59. A Critical Cybersecurity Analysis and Future Research ... https://www.mdpi.com/1424-8220/23/8/4117
60. A Review of EEG-Based Brain-Computer Interfaces as ... https://www.researchgate.net/publication/255705134_A_Review_of_EEG-Based_Brain-Computer_Interfaces_as_Access_Pathways_for_Individuals_with_Severe_Disabilities

61. Protecting Privacy of Users in Brain-Computer Interface ... https://www.researchgate.net/publication/334286648_Protecting_Privacy_of_Users_in_Brain-Computer_Interface_Applications
62. (PDF) Enabling Access for Persons with Visual Impairment https://www.academia.edu/50332851/Enabling_Access_for_Persons_with_Visual_Impairment
63. hw3_stats_google_1gram.txt https://www.cs.cmu.edu/~roni/11761/2017_fall_assignments/hw3_stats_google_1gram.txt
64. Spam Message Detector https://www.kaggle.com/code/dev0914sharma/spam-message-detector
65. Computer Science https://www.arxiv.org/list/cs/new?skip=25&show=1000
66. Klüppelberg, Straub, Welpe - 2014 - Risk A ... https://www.scribd.com/document/907718564/Kluppelberg-Straub-Welpe-2014-Risk-a-Multidisciplinary-Introduction-Unknown
67. Lecture Notes | PDF | Accelerometer | Smartphone https://www.scribd.com/document/910878686/Lecture-Notes
68. (PDF) Focō, Ergo Volō - "I Focus, Therefore I Will" https://www.academia.edu/144259145/Foc%C5%8D_Ergo_Vol%C5%8D_I_Focus_Therefore_I_Will_
69. Decentralized Identifiers (DIDs) v1.0 https://www.w3.org/TR/2021/WD-did-core-20210118/
70. invitation to market design | Oxford Review of Economic Policy https://academic.oup.com/oxrep/article/33/4/541/4587951
71. Usable Security and Verification for Distributed Robotic ... https://search.proquest.com/openview/48398313fdb7cdf6bd5ff1bdc8ce9350/1?pq-origsite=gscholar&cbl=18750&diss=y
72. JPM Big Data and AI Strategies | PDF | Machine Learning https://www.scribd.com/document/369035491/JPM-Big-Data-and-AI-Strategies
73. C O N G R ES S P R O C EED IN G S B O O K https://www.researchgate.net/profile/Bashir-Sanyinna/publication/389298460_KUBA_21-28_OCAK_2025_KITAP/links/67bd9639f5cb8f70d5beb02c/KUeBA-21-28-OCAK-2025-KITAP.pdf?origin=scientificContributions
74. Machine Learning, Image Processing, Network Security and ... https://link.springer.com/content/pdf/10.1007/978-3-031-24367-7.pdf
75. xferlexicon.txt https://www.cs.cmu.edu/afs/cs.cmu.edu/project/cmt-40/Nice/Transfer/Chinese/xferlexicon.txt
76. Chapter 296-880 WAC https://lni.wa.gov/rulemaking-activity/AO19-05/1905Adoption.pdf

77. On the Importance of the Kullback-Leibler Divergence ... https://aclanthology.org/D19-5612/

78. Minimal Kullback-Leibler Divergence for Constrained Lévy- ... https://arxiv.org/pdf/2206.14844

79. The best approach to preventing the value of the Kullback ... https://stackoverflow.com/questions/76566737/the-best-approach-to-preventing-the-value-of-the-kullback-leibler-divergence-bec

80. Backward cloud transformation algorithm based on ... https://pubmed.ncbi.nlm.nih.gov/41592095/

81. Decentralized Identifiers (DIDs) v0.9 https://pr-preview.s3.amazonaws.com/w3c-ccg/did-spec/41/65f5638...c6db2cb.html

82. A Hierarchical Decomposition of Kullback-Leibler Divergence https://arxiv.org/pdf/2504.09029

83. Hw3 Stats Google 1gram | PDF | Internet Forum | Software https://www.scribd.com/document/917428277/Hw3-Stats-Google-1gram

84. Advances in Computing - Springer Link https://link.springer.com/content/pdf/10.1007/978-3-031-75233-9.pdf

85. Ijair Volume 6 Issue 3 IV July September 2019 https://www.scribd.com/document/450712127/ijair-volume-6-issue-3-iv-july-september-2019

86. Published Password Lists: 1 https://ineapple.com/known_pass1

87. HiPEAC Vision 2023 https://inria.hal.science/hal-04023794/file/HiPEAC_Vision_2023.pdf

88. Going Digital in Brazil (EN) https://www.oecd.org/content/dam/oecd/en/publications/reports/2020/10/going-digital-in-brazil_532cb108/e9bf7f8a-en.pdf

89. Apple Academic Press Author Copy https://www.researchgate.net/profile/Antonio-Pimenta-De-Brito/publication/355107226_Brito_A_P_2024_People_Analytics_in_the_COVID-19_Pandemic_how_empathy_and_privacy_turned_out_the_hot_topics_In_MJ_Sousa_S_K_Pani_FD_Mas_S_Sousa_Eds_Advancements_in_Artificial_Intelligence_in_The_Servic/links/64f8990b05a98c1b63f79fca/Brito-A-P-2024-People-Analytics-in-the-COVID-19-Pandemic-how-empathy-and-privacy-turned-out-the-hot-topics-In-MJ-Sousa-S-K-Pani-FD-Mas-S-Sousa-Eds-Advancements-in-Artificial-Intelligence-in-The-S.pdf

90. Instructions for submitting a technical report or thesis. https://cs.nyu.edu/dynamic/reports/?year=all

91. Data Mining.txt https://www.aminer.cn/lab-datasets/crossdomain/Data%20Mining.txt

92. Using const defined in generic trait-bound types in Rust https://stackoverflow.com/questions/78521572/using-const-defined-in-generic-trait-bound-types-in-rust

93. Township of Toms River, NJ Land Use and Development ... https://ecode360.com/11762651

94. NW-C35NW Region HQ Remodel ... - Washington State Parks https://parks.wa.gov/sites/default/files/2024-11/NW-C35NW%20Region%20HQ%20Remodel-Specs.pdf

95. Leveraging zero knowledge proofs for blockchain-based ... https://www.sciencedirect.com/science/article/pii/S2214212623002624

96. (PDF) Implementing zero-knowledge proof authentication ... https://www.researchgate.net/publication/387616317_Implementing_zero-knowledge_proof_authentication_on_Hyperledger_fabric_to_enhance_patient_privacy_and_access_control

97. A Hybrid Continual Learning Framework for Adaptive Fault ... https://www.mdpi.com/2624-831X/7/1/12

98. Intelligibility of Recurrent Neural Networks via Finite State ... https://theses.hal.science/tel-05036659v1/document

99. Statistical Foundations of Data Science - Jianqing Fan https://fan.princeton.edu/document/1226

100. Rebecca Duncan - The Edinburgh Companion To ... https://www.scribd.com/document/791482609/Rebecca-Duncan-The-Edinburgh-Companion-to-Globalgothic-Edinburgh-University-Press-2023

101. Rust generic parameters and compile time if https://stackoverflow.com/questions/73099266/rust-generic-parameters-and-compile-time-if

102. ISSN: 2581 - 8317 https://www.researchgate.net/profile/Sanjay-Singh-173/publication/364537078_Integrated_Input_Management_in_Rainfed_Agro-Ecosystems/links/635223ab96e83c26eb3bd62d/Integrated-Input-Management-in-Rainfed-Agro-Ecosystems.pdf

103. Pedagogical Encounters in the Post-Anthropocene, Volume 2 https://link.springer.com/content/pdf/10.1007/978-3-031-54783-6.pdf

104. (PDF) Psychopathologies of Cognitive Capitalism Part 2 https://www.academia.edu/10801026/Psychopathologies_of_Cognitive_Capitalism_Part_2