# KRILL: Bio-Inspired Network Architecture for the Internet of Things

A Stigmergic, Immunological, and Metabolic Framework for Post-Blockchain Distributed Consensus

**Version 1.0 — February 2026**

**Authors:** KRILL Research Group

---

> *"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it." — Mark Weiser, "The Computer for the 21st Century" (1991)*

## What is KRILL? (Plain-Language Summary)

**The problem:** Billions of IoT devices (sensors, smart home gadgets, industrial monitors) need to share data and trust each other — but blockchain, the go-to technology for "trustless" systems, is too heavy, too slow, and too power-hungry for tiny battery-powered devices with kilobytes of memory.

**Our approach:** Instead of copying financial technology (blockchain), we copied *biology*. Living organisms solved the same problem billions of years ago — trillions of cells coordinate without a central brain, detect intruders (immune system), forget irrelevant information (metabolism), and self-organize into structures (morphogenesis).

**KRILL is a protocol with 9 biological mechanisms adapted for IoT:**

1. **Stigmergic Consensus** — devices leave digital "pheromone trails" (like ants) to agree on sensor readings. No voting rounds needed.
2. **Immune System** — five layers detect anomalies, attackers, and malfunctioning devices. Works like your body's innate + adaptive immunity.
3. **Metabolic State** — old data loses energy and dies (apoptosis). No infinite ledger growth.
4. **Entropic Data Valuation** — the network automatically values data by its information content, not by who sent it.
5. **Quorum Sensing** — devices collectively switch between solo-mode and group-mode based on neighbor density (like bacteria deciding to form a biofilm).
6. **Horizontal Gene Transfer** — firmware updates spread device-to-device like genes between bacteria. No central update server needed.
7. **Morphogenetic Topology** — the network self-organizes its shape using reaction-diffusion patterns (like a developing embryo).
8. **Thymic Tolerance** — the system learns to distinguish "self" from "other" to prevent false alarms.
9. **Immunological Memory** — once the network detects a threat, it remembers and can respond faster next time (like vaccination).

**Result:** An IoT network that uses ~1000x less energy than blockchain, works on $2 microcontrollers, tolerates intermittent

connectivity, automatically heals, and doesn't need any central server.

**Status:** This is a research paper describing the theory and mathematics. A companion *Engineering Specification* document provides byte-level implementation details. No running code exists yet — this is an open invitation to build it.

## Abstract

We present a fundamentally new architectural paradigm for Internet of Things (IoT) distributed systems that departs from blockchain's ledger-centric model in favor of biologically-inspired mechanisms. The proposed framework — designated **KRILL Bio-Inspired Architecture (KRILL-BIA)** — introduces nine interlocking subsystems drawn from evolutionary biology, immunology, swarm intelligence, and developmental biology: (1) **Stigmergic Consensus**, replacing Byzantine Fault Tolerance with pheromone-field coordination grounded in physical-world constraints; (2) a **Pentastratic Immune System** providing multi-layered anomaly detection with autoimmune tolerance; (3) **Metabolic State** with thermodynamically-inspired information decay and programmed apoptosis; (4) **Entropic Data Valuation** using conditional mutual information for autonomous resource allocation; (5) **Quorum Sensing** for leaderless mode selection; (6) **Horizontal Gene Transfer** for decentralized firmware evolution; (7) **Morphogenetic Topology** via reaction-diffusion dynamics; (8) **Thymic Tolerance Training** for false-positive suppression; and (9) **Immunological Memory** with stigmergic vaccine propagation.

We provide mathematical formalizations, energy complexity analysis, honest risk assessment, and comparison with existing distributed systems literature. Three of the proposed mechanisms — physical-world consensus grounding, horizontal gene transfer for firmware, and reaction-diffusion network topology — represent genuinely novel contributions with no direct precedent in published literature. The remaining six combine existing concepts in an unprecedented integrated framework that transforms isolated techniques into a coherent bio-mimetic system.

**Keywords:** Internet of Things, Stigmergy, Artificial Immune Systems, Swarm Intelligence, Distributed Consensus, Reaction-Diffusion Systems, Information Theory, Bio-Inspired Computing, Post-Blockchain Architecture

## Table of Contents



---

## 1. Introduction and Motivation

### 1.1 The Scale of the Problem

The Internet of Things (IoT) is projected to encompass over 30 billion connected devices by 2030 (Statista, 2024), generating an estimated 80 zettabytes of data annually (IDC, 2023). These devices span a hardware spectrum from sub-dollar microcontrollers with 32KB of RAM to autonomous vehicles with multi-core processors. They operate under constraints fundamentally unlike those of traditional computing nodes: intermittent connectivity, energy harvesting, physical exposure to adversarial environments, and deployment lifetimes measured in decades.

Despite this scale, the dominant paradigm for trustless coordination among distributed agents — blockchain — was designed for an entirely different domain. Bitcoin (Nakamoto, 2008) and its successors model the problem of **financial consensus**: how do parties who don't trust each other agree on account balances? This is a legitimate and important problem, but it is not the IoT problem.

### 1.2 The Impedance Mismatch

The IoT problem is not "Who has how much money?" but rather "What is the physical state of the world, and can we trust the entities reporting it?" This distinction has profound architectural consequences:

| DIMENSION | FINANCIAL CONSENSUS (BLOCKCHAIN) | PHYSICAL CONSENSUS (IOT) |
|---|---|---|
| **Nature of truth** | Social construct (balances are agreed upon) | Physical reality (temperature exists independently of measurement) |
| **Verification** | Computational (validate signature + state transition) | Physical (cross-reference with laws of physics) |
| **Cost of lying** | Economic (stake slashing) | Physical + Economic (must defeat physics AND economics) |
| **State growth** | Monotonic (ledger only appends) | Cyclical (old measurements lose relevance) |
| **Participant resources** | Abundant (servers, bandwidth) | Scarce (microcontrollers, batteries) |
| **Communication** | Always-on, low-latency | Intermittent, high-latency, NAT'd |
| **Consensus overhead** | Acceptable (tens of validators) | Prohibitive (billions of participants) |

This impedance mismatch is not a matter of optimization — it is categorical. Blockchain is the wrong *metaphor* for IoT, not merely the wrong *implementation*.

### 1.3 The Biological Insight

We observe that biology solved an isomorphic problem 3.8 billion years ago. A multicellular organism is a distributed system of ~37 trillion cells (Bianconi et al., 2013) that must:

- Coordinate behavior without central control
- Detect and neutralize malicious agents (pathogens) among trusted agents (own cells)
- Manage state (cell differentiation, protein expression) under strict energy constraints
- Self-organize topology (morphogenesis) from simple local rules

- Evolve and adapt to novel threats without redesigning the whole system
- Handle component failure gracefully (cell death → regeneration)

Biology's solutions are not merely "good enough" — they are, in many cases, information-theoretically optimal under energy constraints (Laughlin, 2001; Bialek, 2012). This paper argues that these solutions can be formalized, adapted, and applied to IoT distributed systems, yielding a framework that is fundamentally superior to blockchain for physical-world coordination.

### 1.4 Scope and Claims

This paper makes the following contributions:

1. **Formal definition of stigmergic consensus** with physical-world grounding — a mechanism that eliminates the need for Byzantine agreement for the majority of IoT interactions involving continuous physical quantities (Section 3). We estimate this covers >90% of IoT sensor traffic based on the prevalence of continuous measurements (temperature, humidity, pressure, acceleration) versus discrete events (door open/close, button press); the exact fraction depends on deployment composition.
2. **A five-layer artificial immune system** that goes beyond existing AIS literature (Forrest et al., 1994; de Castro & Timmis, 2002) by incorporating autoimmune tolerance training (Section 4)
3. **Metabolic state management** with thermodynamically-motivated information decay and programmed apoptosis — a concept with no direct precedent in distributed systems (Section 5)
4. **Entropic data valuation** using conditional mutual information for autonomous incentive alignment (Section 6)
5. **Quorum sensing** adapted from bacterial signaling for leaderless mode selection (Section 7)
6. **Horizontal gene transfer** for decentralized firmware propagation — unprecedented in distributed systems literature (Section 8)
7. **Morphogenetic topology** using Turing's reaction-diffusion equations for self-organizing network structure — unprecedented in network architecture (Section 9)
8. **Energy complexity analysis** demonstrating $10^3$-$10^7\times$ improvement over BFT consensus at IoT scale (Section 11)
9. **Honest risk assessment** identifying five fundamental limitations and five open research problems (Section 12)

We explicitly note what is genuinely novel, what is adaptation of existing work, and what remains unproven.

## 2. Foundational Thesis: Why Blockchain is the Wrong Metaphor

### 2.1 The Ontological Argument

Blockchain's conceptual foundation is a **ledger** — a record of agreements between parties. In a ledger, truth is constituted by consensus: the balance in your account is whatever the majority of validators agree it is. There is no external reality against which to verify the claim "Alice has 10 ETH." The claim *is* the consensus.

This ontology is appropriate for finance. It is inappropriate for physics.

When a temperature sensor reports 23°C, there *is* an external reality: the actual temperature of the physical medium. Two honest sensors in the same location, at the same time, measuring the same physical quantity, will agree — not because they reach consensus, but because they measure the same reality. Their agreement is a consequence of physics, not of protocol.

**Definition 2.1 (Physical Ground Truth).** A measurement domain D possesses physical ground truth if and only if there exists an objective physical quantity Q(x, t) at position x and time t such that any correctly functioning sensor s_i at position x_i and time t_i produces a reading q_i satisfying:

```
|q_i - Q(x_i, t_i)| ≤ σ_i
```

where σ_i is the manufacturer-specified measurement uncertainty of sensor s_i.

**Observation 2.1.** IoT sensor data overwhelmingly possesses physical ground truth. Temperature, humidity, pressure, acceleration, voltage, light intensity, chemical concentration — these are objective physical quantities. In contrast, financial balances, smart contract states, and token ownership possess *no* physical ground truth; they exist only as social agreements.

### 2.2 The Complexity Argument

Byzantine Fault Tolerance (BFT), in its classical formulation (Lamport, Shostak & Pease, 1982), requires $O(n^2)$ messages per consensus round for n validators. Practical BFT variants (PBFT: Castro & Liskov, 1999; HotStuff: Yin et al., 2019) reduce this to O(n) messages per round with a

leader-based protocol, but still require all validators to participate in every round.

For a network of $N = 10^9$ IoT devices, even with sharding into groups of $k = 100$ validators, the total communication cost per consensus round is:

```
C_BFT = (N/k) × O(k) = O(N) messages globally
```

At one consensus round per second, this is $10^9$ messages per second — **the equivalent of the entire global internet traffic devoted to consensus alone** (Cisco, 2023: global internet traffic ≈ $5 \times 10^9$ packets/second).

This is not merely expensive. It is physically impossible.

### 2.3 The Thermodynamic Argument

Blockchain maintains state indefinitely. The Ethereum state trie has grown to over 400 GB (Etherscan, 2025) and increases monotonically. This violates a fundamental principle of information physics:

**Landauer's Principle** (Landauer, 1961): Maintaining a bit of information in a non-equilibrium state requires a minimum energy expenditure of $kT \ln 2$ per unit time, where k is Boltzmann's constant and T is temperature.

A system that accumulates state without bound requires unbounded energy to maintain that state. Blockchain circumvents this by distributing the maintenance cost across an ever-growing set of full nodes — but the cost exists and grows monotonically.

In biological systems, information maintenance is tightly coupled to energy availability. Cells that cannot maintain their internal state undergo **apoptosis** (programmed cell death). Information that is no longer referenced degrades naturally. Memory consolidation compresses detailed short-term memories into schematic long-term representations (McClelland, McNaughton & O'Reilly, 1995).

**Thesis 2.1.** A distributed system for IoT should model state as a thermodynamic quantity — subject to decay, requiring energy to maintain, and undergoing natural compression over time. This is not a limitation but a feature: old sensor readings *genuinely are* less valuable than new ones.

### 2.4 The Proposed Alternative: Network as Organism

We propose treating the IoT network not as a ledger but as a **multicellular organism**:

| BLOCKCHAIN CONCEPT | BIOLOGICAL EQUIVALENT | KRILL-BIA IMPLEMENTATION |
|---|---|---|
| Consensus round | — (unnecessary for physics) | Stigmergic pheromone field |
| Reputation score | — (too simplistic) | Pentastratic immune system |

| BLOCKCHAIN CONCEPT | BIOLOGICAL EQUIVALENT | KRILL-BIA IMPLEMENTATION |
|---|---|---|
| State storage | DNA, epigenetics, memory | Metabolic state with half-life |
| Transaction fee | — (wrong incentive) | Entropic data valuation |
| Leader election | — (centralization risk) | Quorum sensing |
| Firmware update (centralized) | Horizontal gene transfer | Decentralized module propagation |
| Network topology (designed) | Morphogenesis | Reaction-diffusion self-organization |
| Slashing (punishment) | Apoptosis (self-destruction) | Programmed state/device death |
| Block confirmation | Immune clearance | Physical consistency verification |

This is not metaphor for its own sake. Each biological mechanism maps to a specific, formalizable computational primitive that solves a specific problem more efficiently than its blockchain counterpart.

---

## 3. Stigmergic Consensus: The End of Byzantine Agreement

### 3.1 Definition and Biological Origin

**Stigmergy** (from Greek στίγμα "mark" + ἔργον "work") was first described by Grassé (1959) in the context of termite mound construction. Termites do not communicate directly about construction plans. Instead, each termite deposits material (pheromone-laced mud) in the environment. Other termites respond to the deposited material, creating complex structures through indirect coordination.

**Definition 3.1 (Stigmergic Coordination).** A set of agents $\{a_1, \ldots, a_n\}$ achieves stigmergic coordination over a shared environment E if:

1. Each agent a_i modifies E locally: E ← E + δ_i
2. Each agent reads E locally: a_i observes E(x_i, t)
3. Agent behavior is a function of local environment state: action(a_i) = f(E(x_i, t))
4. Global coordination emerges without direct inter-agent communication

The key insight is that the *environment itself* serves as the communication medium. In IoT, the physical world plays this role.

### 3.2 The Physical Consistency Constraint

**Theorem 3.1 (Physical Consistency Bound).** For two sensors s_i and s_j measuring the same physical quantity Q, with positions x_i and x_j, measurement times t_i and t_j, and manufacturer uncertainties σ_i and σ_j, the maximum physically permissible difference between their readings is:

```
|q_i - q_j| ≤ ε(d(x_i, x_j), |t_i - t_j|) + σ_i + σ_j
```

where:

- **d(x_i, x_j)** is the Euclidean distance between sensors
- **ε(d, Δt)** is the maximum physically possible difference, bounded by:
  - Spatial gradient: ε_spatial = ∇Q_max · d (maximum gradient of Q times distance)
  - Temporal change: ε_temporal = (∂Q/∂t)_max · Δt (maximum rate of change times time difference)
  - Combined: ε(d, Δt) = ε_spatial + ε_temporal
- **σ_i, σ_j** are measurement uncertainties (from manufacturer datasheets)

**Proof sketch.** Q is a physical field governed by conservation laws. Its spatial gradient is bounded by the maximum energy flux in the medium (e.g., for temperature: thermal conductivity limits how fast gradients can develop). Its temporal derivative is bounded by the maximum energy input/output rate. These bounds are obtainable from physical constants and environmental parameters. □

**Corollary 3.1.** If the physical consistency bound is satisfied, the readings are consistent with physical reality. No consensus protocol is needed to verify this — the laws of physics guarantee it.

**Corollary 3.2.** If the physical consistency bound is violated, at least one sensor is either (a) malfunctioning, (b) compromised, or (c) measuring a genuine anomaly (e.g., a fire). This is the ONLY case where further verification is required.

### 3.3 The Pheromone Field

We define a **pheromone field** P(x, t) over the physical deployment space:

**Definition 3.2 (Pheromone Field).** Each device i at position x_i, at time t, with immunological weight w_i (see Section 4), deposits a pheromone trace:

```
P_i(x, t) = w_i · δ(x - x_i) · e^{-λ(t - t_i)}
```

where:

- w_i ∈ [0, 1] is the immunological health weight of device i

- δ(·) is the Dirac delta function (localized at the device position)
- λ > 0 is the pheromone decay rate (biological half-life: T½ = ln(2)/λ)
- t_i is the time of the last trace deposit

The **global pheromone field** is the superposition:

```
P(x, t) = Σᵢ P_i(x, t) = Σᵢ w_i · δ(x - x_i) · e^{-λ(t - t_i)}
```

In practice, P is implemented as entries in a local Directed Acyclic Graph (DAG). A device does not "send a transaction to the network." It deposits a trace in the local DAG. Neighboring nodes read traces and add their own. Traces naturally decay (are garbage-collected after their half-life elapses).

### 3.4 The Stigmergic Protocol

```
NORMAL OPERATION (majority of interactions — for continuous physical
quantities):
  1. Device i measures physical quantity: q_i = sense()
  2. Device deposits trace in local DAG: DAG.append(q_i, sig_i, t_i)
  3. Neighboring devices read trace
  4. Physical consistency check:
     FOR each neighbor j:
       IF |q_i - q_j| ≤ ε(d_ij, Δt_ij) + σ_i + σ_j:
         PASS (physically consistent — no action needed)
       ELSE:
         TRIGGER immune response (Section 4)
  5. Trace decays naturally: after T½, trace weight → 0.5 · original

COST: O(k) where k = number of neighbors (typically 5-20)
GLOBAL COMMUNICATION: ZERO
CONSENSUS ROUNDS: ZERO
```

### 3.5 Comparison with Byzantine Agreement

| PROPERTY | CLASSICAL BFT (E.G., HOTSTUFF) | STIGMERGIC CONSENSUS |
|---|---|---|
| Messages per round | O(n) per shard | O(k) per device, k << n |
| Global state | Required (all validators agree) | Not required (local fields) |
| Assumption | >2/3 honest validators | >2/3 honest sensors + physical reality exists |
| Verification | Computational (signature checking) | Physical (consistency with laws of physics) |
| Cost at $n = 10^9$ | Physically impossible | $O(k \cdot 10^9) = O(10^{10})$ local operations (feasible) |
| Handles physical anomalies | Treats anomaly as disagreement | Treats anomaly as immune event |

### 3.6 Attack Analysis: Pheromone Poisoning

**Attack vector:** An adversary floods the local DAG with false traces to corrupt the pheromone field.

**Defense (three layers):**

1. **Immunological weight filtering.** Traces from unknown or low-health devices have w_i ≈ 0, contributing negligibly to the field. An attacker must first build immunological credibility (Section 4.4: Thymic Tolerance Training requires 7-30 days).
2. **Physical consistency rejection.** False traces that violate the physical consistency bound (Theorem 3.1) are automatically detected and rejected. The attacker must produce readings that are physically plausible — i.e., consistent with actual physical conditions. This requires the attacker to either (a) know the actual physical state (negating the purpose of lying) or (b) place physical sensors near the target (incurring real-world cost).
3. **PUF-bound identity.** Device identity is cryptographically bound to a Physical Unclonable Function (PUF), typically derived from SRAM power-on patterns (Guajardo et al., 2007). PUF responses cannot be cloned or predicted, making Sybil attacks require physical fabrication of hardware.

**Cost-of-attack analysis:** To poison a pheromone field, the attacker must:

- Deploy physical hardware in physical proximity to the target ($$$ + logistics)
- Maintain hardware through thymic tolerance period (7-30 days)
- Produce physically consistent readings (requires knowing or measuring actual conditions)
- Sustain the attack continuously (traces decay; attack effect is temporary)

This attack profile is fundamentally more expensive than network-based attacks against BFT systems, where the attacker needs only compute and bandwidth.

### 3.7 Clock Synchronization and Temporal Grounding

Pheromone decay ( `e^{-λ(t-t_i)}` ) and metabolic half-life ( `2^{-(t-t₀)/T½}` ) both depend on time. This raises a fundamental question: **how do nodes agree on time?**

**The problem is real but bounded.** Unlike financial consensus, KRILL-BIA does not require nanosecond precision. Pheromone half-lives are measured in minutes; metabolic half-lives in hours to months. Clock errors of seconds — even minutes — are tolerable.

**Three-tier clock strategy:**

```
TIER 1: NANO/FULL NODES (have internet connectivity)
  Clock source: NTP or GPS (when available)
```

```
  Accuracy: ~1-50 ms
  These nodes serve as time anchors for local clusters

TIER 2: DUST NODES WITH RTC (Real-Time Clock chip)
  Clock source: RTC crystal, periodically synchronized with Tier 1
neighbors
  Accuracy: ~1-10 seconds drift per day
  Sync protocol: piggyback timestamp on every pheromone trace exchange
    → each trace carries sender's timestamp
    → receiver computes offset: Δt = t_received - t_claimed - RTT/2
    → Kalman filter over offsets from multiple neighbors → robust local
time

TIER 3: DUST NODES WITHOUT RTC (cheapest MCUs)
  Clock source: internal RC oscillator only (±1-5% frequency error)
  Accuracy: minutes of drift per day if unsynchronized
  Sync protocol: RELATIVE TIME ONLY
    → device does not maintain absolute time
    → pheromone traces timestamped with MONOTONIC COUNTER
    → neighbors translate counter to real time using observed message
intervals
    → decay computed by NEIGHBORS, not by the device itself
    → device reports: "this reading is 47 ticks after my last reading"
    → neighbors know tick duration from calibration (observed over neonatal
period)
```

**Theorem 3.2 (Temporal Consistency Bound).** For pheromone decay to remain meaningful, the clock error δt between two nodes must satisfy:

```
δt << T½_min / ln(2)
```

where T½_min is the shortest half-life in use (EPHEMERAL state, T½ ≈ 10 minutes). This yields δt << 14.4 minutes. Even the worst Tier 3 node (5% oscillator drift) accumulates only ~72 seconds of error per day, well within this bound if synchronized at least once per day.

**Failure mode:** If a Dust node is completely isolated (no neighbors) for extended periods, its local time becomes meaningless. This is handled by Quorum Sensing (Section 7): isolated nodes enter SOLO mode, where pheromone decay is suspended and data is stored with monotonic counters only. Upon reconnection, neighbors reconstruct the temporal ordering from counter sequences and known inter-message intervals.

**Key insight:** Biological organisms do not have synchronized clocks either. Circadian rhythms are entrained by external signals (light, temperature) and drift when isolated. KRILL-BIA's approach is identical: time is entrained by neighbor interactions, not by global synchronization.

### 3.8 Boundary Environments and Spatial Discontinuities

**Problem:** The physical consistency bound (Theorem 3.1) assumes smooth spatial gradients. At environmental boundaries — a wall between indoor/outdoor, the edge of a furnace, the boundary of a cold room — two

sensors 1 meter apart may experience a 30°C difference. The consistency bound would flag both as anomalous.

**Solution: Boundary-Aware Consistency**

```
APPROACH 1: Learned gradient profiles
  During neonatal period, the network observes maximum inter-neighbor
differences.
  If sensors A and B consistently differ by 25°C ± 3°C, this is stored as
  a BOUNDARY PROFILE:
    boundary(A, B) = {mean_diff: 25°C, std_diff: 3°C}

  Physical consistency check becomes:
    |q_A - q_B - boundary(A,B).mean_diff| ≤ ε(d, Δt) + σ_A + σ_B +
2·boundary(A,B).std_diff

  The bound ADAPTS to known spatial discontinuities.

APPROACH 2: Multi-modality cross-check
  At boundaries, different physical quantities may still correlate:
    - Indoor sensor: 22°C, 40% RH, 1013 hPa
    - Outdoor sensor: -5°C, 80% RH, 1013 hPa

  Temperature disagrees → boundary violation
  Pressure agrees → both sensors functional
  Humidity correlates with temperature → physically consistent

  Cross-modality verification resolves boundary ambiguity.

APPROACH 3: Explicit boundary declaration
  Deployment metadata includes known boundaries.
  Operators declare: "Sensor A is inside, Sensor B is outside."
  Consistency check uses appropriate physics for each environment
  independently, rather than cross-checking across the boundary.

  This is the least elegant but most robust approach.
```

**Residual risk:** Undeclared, unlearned boundaries during the first deployment period may cause transient false positives. The thymic tolerance mechanism (Section 4.5) absorbs these: during the neonatal period, boundary profiles are learned automatically.

### 3.9 Operation on Low-Power Wide-Area Networks (LPWAN)

The preceding sections assume local mesh networking (BLE, WiFi, 802.15.4) with sub-second latency and multi-kilobyte payloads. Many real IoT deployments use **LPWAN** technologies (LoRaWAN, NB-IoT, Sigfox) with fundamentally different constraints:

```
LPWAN CONSTRAINTS vs. MESH:
  Latency:        seconds to minutes (vs. milliseconds)
  Payload:        12-250 bytes per message (vs. kilobytes)
  Duty cycle:     1% (LoRa EU) → max 36 seconds TX per hour
  Topology:       star-of-stars (gateway-centric, not peer-to-peer)
  Bidirectional:  limited (downlink windows are scarce)
```

**Impact on BIA mechanisms:**

| MECHANISM | IMPACT | ADAPTATION |
|---|---|---|
| **Stigmergic trace deposit** | Cannot maintain local DAG with 12-byte payloads | **Compressed trace**: 8-byte payload = {device_id_hash(2B), reading(2B), timestamp_delta(2B), confidence(1B), checksum(1B)}. Gateway reconstructs full trace and maintains DAG on behalf of Dust nodes. |
| **Pheromone decay** | Infrequent updates (minutes apart) → stale field | **Adaptive λ**: decay rate λ is scaled by reporting interval. For LoRa (1 msg/15 min), λ_lora = λ_mesh / 900. Pheromone persists longer because updates are rarer. |
| **Physical consistency check** | Cannot query k neighbors directly (no peer-to-peer) | **Gateway-mediated check**: gateway receives traces from all devices in range, performs consistency check centrally, and flags anomalies. This sacrifices decentralization but preserves physical grounding. |
| **Immune system** | Behavioral fingerprint requires neighbor storage | **Gateway as immune proxy**: gateway stores profiles for LPWAN devices. Strata 0-1 run at gateway; device runs only innate rules locally. |
| **Quorum sensing** | Density detection requires local broadcast | **Implicit quorum**: gateway knows how many devices report per epoch → can compute local density and broadcast mode selection in downlink window. |
| **HGT** | Firmware transfer impossible in 12-byte payloads | **HGT disabled for LPWAN Dust**. Firmware updates via gateway OTA (centralized). Only mesh-connected Nano+ nodes participate in HGT. |

**Architectural consequence:** LPWAN devices operate as **degraded Dust nodes** — they contribute traces and receive innate protection, but rely on gateways for immune, consistency, and topology functions. This is architecturally honest: a LoRa sensor with 1% duty cycle and 12-byte payloads *cannot* participate in decentralized consensus. The gateway acts as a **BIA proxy**, not a trust anchor — physical consistency is still verified against physics, not against the gateway's authority.

**Biological analogy:** Red blood cells in biology lack nuclei and cannot participate in immune responses. They perform their core function (oxygen transport ≈ data transport) and rely on the surrounding immune system

(white blood cells ≈ gateway) for protection. LPWAN devices are the red blood cells of KRILL-BIA.

---

## 4. The Pentastratic Immune System

### 4.1 Critique of Existing Approaches

Current distributed systems employ one-dimensional reputation scores for node trustworthiness. This approach has four fundamental flaws:

1. **Dimensionality collapse.** A scalar reputation cannot distinguish "sensor is miscalibrated" from "sensor is under active attack" from "sensor is in an unusual environment." Each case requires a different response; a single number conflates them.
2. **Global knowledge requirement.** Who aggregates the score? A centralized aggregator is a single point of failure. Distributed aggregation requires consensus — recreating the very problem we're trying to solve.
3. **Irreversibility.** In practice, reputation recovery is prohibitively slow. A sensor that was briefly miscalibrated but is now fixed remains "untrusted" for an extended period, reducing network coverage.
4. **No autoimmune protection.** There is no mechanism to prevent the reputation system from being *too aggressive* — flagging healthy sensors as malicious due to environmental changes (seasonal variation, hardware aging, relocated deployment).

The biological immune system solves all four problems. We adapt its architecture to a five-layer system (hence "pentastratic," from Greek πέντε "five" + Latin stratum "layer"). Our approach builds on the Danger Theory (Matzinger, 2002), which argues that the immune system responds not to "non-self" per se but to *danger signals* — an insight that directly informs our graduated response model (§4.4) where severity of response is proportional to assessed threat level, not merely to foreignness.

### 4.2 Stratum 0: Innate Immunity

**Biological analog:** Neutrophils — first responders, fast, non-specific.

**Implementation:** Hardcoded boundary checks requiring zero computation, zero network access, zero learning.

```
INNATE RULES (examples for temperature sensor):
  IF temperature ∉ [-273.15°C, +10000°C]:   → REJECT (physically
impossible)
  IF temperature ∉ [-40°C, +85°C]:          → REJECT (outside sensor
operating range)
  IF Δtemperature > 50°C per second:         → REJECT (physically
```

```
implausible rate of chang
e)
  IF reporting frequency > 1000 readings/s:   → THROTTLE (hardware cannot
sample this fast)
  IF signature verification fails:            → REJECT (cryptographic
failure)
```

**Properties:**

- Execution time: O(1), < 1µs
- Energy cost: negligible
- False positive rate: effectively 0% (bounds are generous)
- Detection scope: only catches gross violations
- No learning, no state, no network dependency

This is the BIOS of the immune system. It stops obvious garbage before any resources are wasted on sophisticated analysis.

### 4.3 Stratum 1: Behavioral Fingerprint (Metabolic Profile)

**Biological analog:** Major Histocompatibility Complex (MHC) — the molecular identity card that immune cells use to distinguish "self" from "other."

**Implementation:** Each device has a multi-dimensional behavioral profile, but — critically — **the device does not know its own profile.** The profile is maintained by neighbors, exactly as in biology: a cell does not evaluate its own normality; the immune system does.

**Profile vector** for device i:

```
Profile(i) = {
  μ_values:     distribution of measurement values (histogram, 50 bins)
  μ_temporal:   activity pattern (when active, when asleep, period)
  μ_comm:       communication pattern (neighbors contacted, frequency)
  μ_energy:     energy signature (consumption vs. activity correlation)
  μ_correlation: cross-correlation with k nearest neighbors
}
```

Each component is a distribution, not a scalar. The profile captures the device's **normal behavior in context** — including its correlations with the physical environment.

**Storage:** Each neighbor stores approximately 500 bytes of profile per monitored device. For k = 10 neighbors monitoring each device, total profile storage per device is 5 KB distributed across the network.

### 4.4 Stratum 2: Adaptive Immunity (Anomaly Detection)

**Biological analog:** T-cells — detect specific anomalies, mount proportional response.

**Implementation:** Response is **graduated and proportional**, not binary:

```
ANOMALY CLASSIFICATION:

Deviation < 1σ from profile:
  Classification: NORMAL
  Response: None
  Analogy: Healthy vital signs

Deviation 1σ – 2σ from profile:
  Classification: PYREXIA (elevated temperature)
  Response:
    - Increase monitoring frequency (2×)
    - Log deviation pattern
    - NO penalty. NO isolation. NO weight reduction.
    - Device MAY BE RIGHT — environment may have changed.
  Analogy: Low-grade fever. Watch and wait.

Deviation 2σ – 3σ from profile:
  Classification: INFLAMMATION
  Response:
    - Active cross-verification with neighbors
    - IF neighbors CONFIRM the change (their readings also shifted):
        → UPDATE profile (adaptive learning)
        → Reclassify as new normal
    - IF neighbors DENY the change (their readings are stable):
        → Escalate to Stratum 3
  Analogy: Significant fever. Run tests.

Deviation > 3σ AND no neighbor confirmation:
  Classification: REJECTION
  Response:
    - Device isolated from consensus participation
    - Data placed in quarantine (NOT deleted — may be valuable)
    - Attempt automated remediation (restart, recalibration)
    - If remediation fails → escalate to human operator
  Analogy: Organ rejection. Serious intervention required.
```

**Key design principle:** The system never assumes an anomalous reading is wrong. It only acts on anomalies that are *inconsistent with neighbors*. A sensor reading that suddenly changes because of a real event (fire, HVAC failure, door opening) will be *confirmed by neighbors* and the profile will adapt.

### 4.5 Stratum 3: Thymic Tolerance (Self/Non-Self Discrimination)

**Biological analog:** Thymic selection — in the thymus, T-cells that react too strongly to self-antigens are eliminated, preventing autoimmune disease (Klein et al., 2014).

**The autoimmune problem in networks:** Without tolerance training, every seasonal change, every hardware drift, every deployment reconfiguration triggers false alarms. The immune system attacks the network it's supposed to protect. This is the distributed-systems equivalent of lupus or multiple sclerosis.

**Implementation:**

```
NEONATAL PERIOD (device joins network):
  Duration: 7-30 days (configurable, default 14 days)

  During this period:
    1. Network OBSERVES device without judging
    2. Neighbors build Profile(i) from raw observations
    3. Device readings have LOWER confidence weight (w_neonatal = 0.3)
    4. No anomaly detection runs against this device
    5. Device participates in data collection but not in immune decisions

  At end of neonatal period:
    1. Profile(i) is FROZEN as "self-signature"
    2. Self-signature stored permanently by neighbors
    3. Device weight promoted to w_normal = 0.8
    4. Anomaly detection begins using self-signature as baseline

  SELF-SIGNATURE is IMMUTABLE for drift detection (Stratum 4).
  PROFILE is MUTABLE (adapts to confirmed environmental changes).

  The distinction is critical:
    - Profile changes with the environment (seasons, upgrades)
    - Self-signature captures the INTRINSIC behavior of the device
    - Drift detection compares against self-signature, not profile
```

**Why this matters:** Without thymic tolerance, a network deployed in summer would flag all devices as anomalous in winter. With it, seasonal changes are absorbed by profile adaptation (confirmed by neighbors), while genuine device degradation is caught by drift detection against the immutable self-signature.

## 4.6 Stratum 4: Immunological Memory

**Biological analog:** Memory B-cells and T-cells — after an infection, the immune system retains a template of the pathogen for rapid response to re-infection. This is the basis of vaccination.

**Implementation (two mechanisms):**

### 4.6.1 Individual Memory (Reinfection Defense)

```
IF device i WAS previously classified as REJECTION and subsequently
recovered:
  → Attack signature stored as "memory cell" by neighbors
  → Upon similar symptoms: IMMEDIATE escalation (skip Strata 0-2)
  → Response time: O(1) instead of O(days)

Memory cell structure:
  {
    signature:   feature vector of the attack (which dimensions deviated,
how)
    timestamp:   when the attack occurred
    resolution:  how it was resolved (restart, recalibration, replacement)
    confidence:  how certain we are this was a real attack (vs. false
positive)
  }
```

### 4.6.2 Vaccination (Preemptive Defense)

```
IF a novel attack type is detected in cluster A:
  → Attack signature packaged as "vaccine"
  → Vaccine propagated STIGMERGICALLY:
      Cluster A neighbors → their neighbors → their neighbors → ...
  → Propagation is LOCAL (not global broadcast)
  → Vaccine strength DECAYS with distance from source:
      vaccine_strength(d) = initial_strength · e^{-d/d₀}
      where d₀ = characteristic propagation distance (~10 hops)
  → Clusters far from the attack receive weak signal (low priority)
  → Clusters near the attack receive strong signal (high priority)

  THIS IS EXACTLY BIOLOGICAL:
    Your immune system doesn't broadcast globally.
    Inflammation signals radiate outward from the infection site,
    mobilizing nearby resources first.
```

### 4.7 The NK Cell Analog: Random Audit

**Biological analog:** Natural Killer (NK) cells — patrol randomly, inspect cells regardless of symptoms, kill cells that fail to present proper MHC markers.

**The slow drift problem (cancer analog):** The most dangerous attack is slow, deliberate drift. A compromised device changes its behavior by 0.1σ per day. After 100 days, it's 10σ from its original behavior but never triggered a 1σ alarm at any single step. This is precisely how cancer evades the immune system.

```
NK CELL MECHANISM:
  Each device has probability p_audit = 0.01 of random full audit per
epoch.

  Audit procedure:
    1. Compare CURRENT behavior against SELF-SIGNATURE (from neonatal
period)
    2. Self-signature is IMMUTABLE — never updated
    3. If total drift > drift_threshold:
        → ALARM (even though no single step exceeded 1σ)
    4. drift_threshold = 5σ from self-signature (calibratable)

  Properties:
    - Expected time to detect slow drift: ~100 epochs (tunable via p_audit)
    - Cost: negligible (0.01 × cost of full comparison)
    - This is the ONLY mechanism requiring LONG-TERM memory
    - All other strata are stateless or short-memory
```

---

## 5. Metabolic State: Information Thermodynamics

### 5.1 The State Bloat Problem

Ethereum's state trie exceeds 400 GB and grows monotonically (Etherscan, 2025). Every smart contract deployed in 2015 still occupies

state, regardless of whether anyone has interacted with it in years. This is the informational equivalent of a city where no building is ever demolished — eventually, the maintenance cost exceeds the value.

The second law of thermodynamics states that maintaining order (low entropy) requires continuous energy expenditure. Blockchain violates this by creating ordered information (state entries) and expecting them to persist indefinitely without ongoing cost. This is, in a thermodynamic sense, a perpetual motion machine of information.

### 5.2 State with Half-Life

**Definition 5.1 (Metabolic State).** Each fragment of state S in KRILL-BIA has an associated confidence function:

```
confidence(S, t) = confidence(S, t₀) · 2^{-(t - t₀) / T½(S)}
```

where T½(S) is the state's **information half-life**, determined by its type.

### 5.3 State Type Taxonomy

**Type I: EPHEMERAL** (T½ = minutes)

```
Examples: individual sensor readings, GPS positions, battery voltage
Rationale: A temperature reading from 2 hours ago is genuinely less
  informative than one from 2 seconds ago. The physical world changes.
After T½: confidence → 0.5 (still present but low-weight)
After 5 × T½: effectively garbage-collected (confidence < 0.03)
```

**Type II: OPERATIONAL** (T½ = days)

```
Examples: smart contract state, device configurations, routing tables
Rationale: Current configuration is relevant; week-old configuration is not
Can be REFRESHED: any interaction resets confidence to 1.0
Refresh COSTS resources (PLANKTON) → maintaining state = ongoing expense
If not refreshed → naturally decays → dead contracts self-destruct
```

**Type III: STRUCTURAL** (T½ = months)

```
Examples: device identity, certificates, cluster membership
Rationale: Identity is persistent but not eternal
Requires PERIODIC HEARTBEAT to maintain (proof of life)
Missing heartbeat → identity decays → "dead" device disappears
No manual cleanup needed — death is automatic
```

**Type IV: ARCHIVAL** (T½ = ∞, but with progressive compression)

```
Examples: aggregated historical data (weekly averages, monthly summaries)
Does NOT decay in confidence — but LOSES RESOLUTION over time:
  Raw minute-by-minute data → (after 1 week) → hourly averages
  Hourly averages → (after 1 month) → daily averages
  Daily averages → (after 1 year) → monthly averages
  Monthly averages → (after 10 years) → yearly averages

THIS IS EXACTLY HOW BIOLOGICAL MEMORY WORKS:
```

```
  You remember yesterday in detail.
  You remember last year schematically.
  You remember childhood in vague impressions.
  The information is progressively COMPRESSED, not deleted.
```

### 5.4 Apoptosis: Programmed Self-Destruction

In biology, apoptosis (Kerr, Wyllie & Currie, 1972) is the mechanism by which a cell detects that it is damaged or no longer needed and initiates its own destruction, for the good of the organism. Apoptosis is not death by external force — it is death by internal decision.

No existing distributed system implements this concept. KRILL-BIA introduces three forms:

#### 5.4.1 Device Apoptosis

```
TRIGGER: Device detects internal compromise
  (firmware integrity check fails, PUF response anomalous,
   secure boot verification fails)

PROTOCOL:
  1. Device broadcasts "death certificate" to neighbors
     (signed with current key: "My identity is no longer trustworthy")
  2. PUF-derived key is rotated (new identity)
  3. Old state → quarantined by neighbors
  4. Device restarts with clean identity
  5. Enters neonatal period again (Stratum 3)

BENEFIT: Self-healing. Compromised devices don't wait for external
  detection — they detect and respond autonomously.
```

#### 5.4.2 State Apoptosis

```
TRIGGER: Smart contract detects internal inconsistency
  (two mutually exclusive conditions both satisfied,
   invariant violated, deadlock detected)

PROTOCOL:
  1. Contract state → "apoptotic" flag
  2. Resources (tokens, bandwidth) → returned to stakeholders
  3. Contract archived with "apoptosis" metadata
  4. Dependent contracts notified
  5. No manual intervention required

BENEFIT: Dead smart contracts don't accumulate. They self-destruct
  when they become inconsistent, freeing resources automatically.
```

#### 5.4.3 Data Apoptosis

```
TRIGGER: Data actively contradicted by newer, higher-confidence data
  (sensor later reports calibration error; firmware update reveals
   measurement bug; physical ground truth proves reading false)

PROTOCOL:
  1. Old data marked as "dead" (not waiting for T½ — immediate)
  2. Downstream consumers notified of data retraction
  3. Hash stub retained (proof that data existed and was retracted)
  4. Full content garbage-collected
```

```
BENEFIT: Incorrect data is actively removed, not just deprioritized.
  The network corrects itself rather than accumulating errors.
```

### 5.5 The Network Life Cycle

The combination of metabolic state and apoptosis creates a natural life cycle:

```
Birth       → Device joins, neonatal period, builds profile
Growth      → Device gains immunological weight, contributes data
Maturity    → Full participation, refreshes state, maintains identity
Aging       → Hardware degrades, profile widens (more noise)
Disease     → Anomaly detected, immune response, treatment or isolation
Death       → Energy depleted / heartbeat missed → state decays, identity
dissolves
Decomposition → Historical data compressed to aggregates
Memory      → Aggregated data persists; individual readings forgotten
```

**Key property:** The network does not grow monotonically. It **breathes**. Old state decays, new state forms. The global state oscillates around an equilibrium determined by network activity — exactly like population dynamics in an ecosystem (Lotka, 1925; Volterra, 1926).

---

## 6. Entropic Data Valuation

### 6.1 Information-Theoretic Foundation

Shannon entropy (Shannon, 1948) quantifies the information content of a signal. For KRILL-BIA, we need a more specific quantity: **conditional mutual information** — how much does a new reading tell us about the world, given what the network already knows?

**Definition 6.1 (Data Value).** The value of a reading r from device i is:

```
V(r, i) = I(r ; World | Network_State) · health(i)
```

where:

- **I(r ; World | Network_State)** is the conditional mutual information between reading r and the physical world state, conditioned on the network's current knowledge
- **health(i)** ∈ **[0, 1]** is the immunological health weight of device i (from Section 4)

### 6.2 Practical Computation

Exact computation of conditional mutual information is intractable for large networks. We use a practical approximation based on local

information:

```
I_approx(r, i) = H(Q_local | N_local) - H(Q_local | N_local, r)
```

where:

- Q_local = physical quantity in device i's neighborhood
- N_local = readings from all neighbors in the last epoch
- H(·|·) = conditional entropy

This can be computed locally using empirical distributions from recent readings.

### 6.3 Scenario Analysis

**Scenario 1: Redundant Reading**

```
Context: 20 sensors report temperature = 23°C ± 0.5°C
New reading: 23.1°C
I(r; World | Network) ≈ 0 (negligible new information)
V(r, i) ≈ 0
PLANKTON reward: minimal (covers transmission cost only)
```

**Scenario 2: First Sensor in Uncovered Area**

```
Context: No existing sensors in this location
New reading: 23.1°C
I(r; World | Network) = H(Q_local) (maximum: total entropy reduction)
V(r, i) = HIGH
PLANKTON reward: maximum
```

**Scenario 3: Anomaly from Healthy Device**

```
Context: 20 sensors report 23°C ± 0.5°C
New reading: 45°C from device with health(i) = 0.95
I(r; World | Network) = VERY HIGH (massive entropy change)
V(r, i) = very high (0.95 × high_info)
Interpretation: possible fire, HVAC failure — high-value alert
PLANKTON reward: very high (device potentially saved the building)
```

**Scenario 4: Anomaly from Unhealthy Device**

```
Context: Same as Scenario 3
New reading: 45°C from device with health(i) = 0.1
V(r, i) = LOW (high_info × 0.1 = low value)
Data → quarantine, not reward
BUT: if neighbors CONFIRM → health(i) retroactively increased
```

### 6.4 Anti-Gaming Properties

**Attempted gaming strategy:** Report random values → high entropy → high reward.

**System response:**

1. Random values violate physical consistency (Theorem 3.1) → immune response
2. health(i) drops rapidly
3. V(r, i) = high_entropy × low_health = LOW reward
4. Continued gaming → full isolation

**Theorem 6.1 (Anti-Gaming).** Gaming the entropic valuation is equivalent to exhibiting anomalous behavior, which triggers the immune system, which reduces health(i), which reduces reward. The immune system and tokenomics form a closed defense loop: any strategy that increases information entropy without genuine information gain simultaneously decreases immunological weight, nullifying the reward.

### 6.5 Emergent Geographic Incentives

The entropic valuation naturally creates economic incentives for optimal sensor placement:

```
Densely covered area (city center):
  I(new_sensor; World | existing_sensors) ≈ 0
  Economic incentive to deploy here: ZERO

Uncovered area (rural field):
  I(first_sensor; World | nothing) = MAXIMUM
  Economic incentive to deploy here: MAXIMUM

EMERGENT RESULT: The network autonomously tends toward
  UNIFORM INFORMATION-THEORETIC COVERAGE.

No central planning required. Entropy does the planning.
```

This is analogous to the **Ideal Free Distribution** in ecology (Fretwell & Lucas, 1969), where organisms distribute themselves across habitats in proportion to resource availability, without central coordination.

---

## 7. Quorum Sensing: Leaderless Collective Decision-Making

### 7.1 Biological Origin

In bacterial colonies, quorum sensing (Fuqua, Winans & Greenberg, 1994) enables population-level behavioral transitions without any centralized decision-maker. Each bacterium continuously secretes small signaling molecules (autoinducers). When the local concentration of autoinducer exceeds a threshold — indicating sufficient population density — all bacteria in the area simultaneously switch behavior (e.g., from planktonic to biofilm formation, or from quiescence to bioluminescence).

No bacterium "decides." The behavior **emerges** from local chemistry.

### 7.2 KRILL-BIA Implementation

Each device periodically emits a **presence pulse** — a minimal network packet (1 byte, zero payload, cryptographically signed). Cost: negligible energy.

The device measures the local density of presence pulses (received per unit time) and adjusts its operational mode accordingly:

```
MODE SELECTION (automatic, no election, no leader):

Density < θ_low (e.g., < 3 neighbors detected):
  MODE: SOLO
  - Device records data locally (flash storage)
  - No cross-verification (insufficient neighbors)
  - Data has LOWER confidence (uncorroborated)
  - Reports to Nano node when connectivity available

θ_low ≤ Density < θ_high (e.g., 3-10 neighbors):
  MODE: QUORUM
  - Cross-verification active
  - Stigmergic DAG maintained among quorum members
  - Physical consistency checked
  - Data has MEDIUM-HIGH confidence

Density ≥ θ_high (e.g., > 10 neighbors):
  MODE: SWARM
  - Full stigmergic field active
  - Immune system operational (all strata)
  - Entropic valuation active
  - Data has MAXIMUM confidence
```

### 7.3 Hysteresis for Mode Stability

A naive threshold-based system oscillates when density fluctuates near the threshold. Biology solves this with **hysteresis** — different thresholds for up-transitions and down-transitions.

**Definition 7.1 (Hysteretic Mode Transition).** Mode transitions follow:

```
Transition UP (e.g., QUORUM → SWARM):
  Requires density > θ_high + Δ (e.g., > 12)

Transition DOWN (e.g., SWARM → QUORUM):
  Requires density < θ_high - Δ (e.g., < 8)

Hysteresis band: [θ_high - Δ, θ_high + Δ] (e.g., [8, 12])
Within band: maintain current mode.
```

This is identical to a thermostat with hysteresis — simple, energy-efficient, and provably stable (no oscillation).

**Theorem 7.1 (Mode Stability).** With hysteresis band width 2Δ, the minimum time between mode transitions is lower-bounded by 2Δ / (maximum rate of density change). For Δ = 2 and a maximum of 1 device

arriving/departing per minute, the minimum inter-transition time is 2 minutes. This prevents rapid oscillation under normal conditions.

---

# 8. Horizontal Gene Transfer: Firmware Without a Center

## 8.1 Biological Origin and Radical Proposal

In bacteria, **horizontal gene transfer** (HGT) enables the transfer of genetic material between organisms that are not in a parent-offspring relationship (Syvanen, 1985; Ochman, Lawrence & Groisman, 2000). A bacterium can acquire antibiotic resistance from a neighboring bacterium of a completely different species. This is fundamentally faster than vertical (hereditary) evolution.

We propose an analogous mechanism for firmware updates in IoT networks. This is, to our knowledge, **unprecedented in distributed systems literature.**

## 8.2 Modular Firmware Architecture

The prerequisite for horizontal transfer is modularity. KRILL-BIA firmware is not a monolithic blob but a set of independently deployable modules:

```
MODULE TAXONOMY:

CORE (≈ genome)
  - Minimal kernel: crypto primitives, network stack, identity management
  - Changed RARELY (months-years)
  - Requires Trust Quorum signature (manufacturer + n-of-m auditors)
  - Cannot be transferred horizontally (too critical)

CAPABILITIES (≈ plasmids)
  - Functional modules: compression algorithms, data parsers, ML models
  - Changed OCCASIONALLY (weeks-months)
  - Requires manufacturer + auditor signature
  - CAN be transferred horizontally between compatible devices

ADAPTATIONS (≈ epigenetics)
  - Local tuning: calibration offsets, sleep schedules, communication
parameters
  - Changed FREQUENTLY (hours-days)
  - Requires device self-signature only
  - Transferred automatically to replacement devices in same deployment
```

## 8.3 Horizontal Transfer Protocol

```
TRANSFER PROTOCOL:

1. ADVERTISEMENT
```

```
   Device A has module M (e.g., "LZ4 compression v2.3")
   A periodically advertises: {module_id, version, signature, size,
compatibility_hash}
   Advertisement is stigmergic — deposited in local DAG, not broadcast

2. EVALUATION (by potential recipient B)
   B checks:
     - Is signature valid? (manufacturer + auditor)
     - Is my hardware compatible? (compatibility_hash matches)
     - Do I have sufficient energy for installation?
     - Is module newer than what I have?
   If all YES → B requests transfer from A

3. TRANSFER
   Direct peer-to-peer link (BLE, WiFi Direct, or local mesh)
   Module transferred with integrity verification (hash check)
   Total transfer: typically 1-50 KB

4. NEONATAL SANDBOX
   New module runs in SANDBOX for 24 hours
   Sandbox: module can execute but cannot modify core state,
     cannot access cryptographic keys, cannot modify other modules
   Performance metrics collected:
     - Energy consumption change
     - Compression ratio change
     - Error rate
     - Stability (crashes, exceptions)

5. ACCEPTANCE OR REJECTION
   After sandbox period:
     IF performance_improved AND no_crashes:
       → Module promoted to production
       → Device reports: "Module M works well" (fitness signal)
     IF performance_degraded OR crashes:
       → Module rejected
       → Automatic rollback to previous version
       → Device reports: "Module M caused problems" (negative fitness)
```

## 8.4 Natural Selection of Modules

Module **fitness** is computed from aggregated field reports:

```
fitness(M) = (devices_reporting_improvement) / (devices_that_installed_M)
```

```
Evolution dynamics:

Module M installed on 100 devices:
  85 report improvement → fitness = 0.85 → module SPREADS
  Neighboring devices see high fitness → install M → spreads further

Module N installed on 50 devices:
  40 report crashes → fitness = 0.20 → module DIES
  Negative fitness signals propagate → no one installs N
  Devices with N → automatic rollback

THIS IS NATURAL SELECTION applied to software:
  - Variation: different modules with different implementations
  - Selection: fitness based on real-world performance
  - Inheritance: high-fitness modules spread to new devices
  - Extinction: low-fitness modules disappear from the population
```

### 8.5 Viral Defense (Malicious Module Protection)

```
DEFENSE LAYERS:

Layer 1: Cryptographic signature requirement
  Module MUST be signed by manufacturer + auditor
  Unsigned modules → rejected immediately
  Forged signatures → cryptographic verification fails → rejected

Layer 2: Neonatal sandbox
  Even with valid signature, module is sandboxed for 24h
  Malicious behavior in sandbox → rejected before it can do damage
  Zero access to keys or core state during sandbox

Layer 3: Fitness-based extinction
  If malicious module passes sandbox but causes long-term harm:
    → Negative fitness reports from victims
    → Fitness drops → module stops spreading
    → Devices with module → auto-rollback

  ATTACK WINDOW: time between deployment and fitness signal accumulation
  Mitigation: conservative deployment (module spreads to max 10% of
    compatible devices before broader rollout, similar to canary
deployment)

Layer 4: Signature revocation
  If auditor's key is compromised → revocation broadcast
  All modules signed by compromised auditor → quarantined
  Devices → rollback to last known-good version
```

---

## 9. Morphogenetic Topology: Self-Organizing Network Structure

### 9.1 Turing's Reaction-Diffusion Model

In his seminal paper "The Chemical Basis of Morphogenesis," Alan Turing (1952) demonstrated that two chemicals diffusing at different rates can spontaneously create stable spatial patterns (spots, stripes, spirals) from an initially homogeneous state. This **reaction-diffusion** mechanism explains zebra stripes, leopard spots, and fingerprint patterns (Murray, 2003; Kondo & Miura, 2010).

The mathematical formulation:

```
∂A/∂t = D_A ∇²A + f(A, I)    (activator)
∂I/∂t = D_I ∇²I + g(A, I)    (inhibitor)

where:
  A = activator concentration
  I = inhibitor concentration
  D_A = activator diffusion coefficient
  D_I = inhibitor diffusion coefficient (D_I > D_A)
  f, g = reaction kinetics
```

The key condition for pattern formation is **D_I > D_A**: the inhibitor diffuses faster than the activator. This creates "local activation, long-range inhibition" — a principle that generates self-organizing spatial patterns.

### 9.2 Application to Network Topology

**This application of reaction-diffusion to network topology is, to our knowledge, unprecedented.**

We define two virtual signals in the network. Note: to satisfy Turing's instability condition (D_I > D_A), the **inhibitor must diffuse faster** than the activator. This is counterintuitive but essential — it creates the "local activation, long-range inhibition" pattern that produces stable spatial structures.

**Activator A(x, t):** "More connectivity is needed here"

- Generated when: data packets are dropped, latency exceeds threshold, coverage gaps detected, new devices join
- Diffusion: **SLOW** (D_A small — stays local, builds up near the source of the need)
- Effect: devices that sense high A increase their communication range, wake up more frequently, relay more aggressively
- Biological analog: morphogen produced at wound site, concentrated locally

**Inhibitor I(x, t):** "Capacity is saturated — neighboring areas should not also expand"

- Generated when: bandwidth saturated, energy reserves depleted, redundant coverage detected
- Diffusion: **FAST** (D_I > D_A — spreads rapidly to wider neighborhood, preventing over-recruitment)
- Effect: devices in the wider neighborhood that sense high I do not increase their activity, preventing global over-reaction
- Biological analog: long-range inhibitory morphogen (e.g., BMP signaling in limb development)

**Key consequence of D_I > D_A:** When a coverage gap appears, the activator builds up *locally* at the gap, recruiting nearby devices to compensate. Simultaneously, the inhibitor spreads *far beyond* the gap, preventing the entire region from over-reacting. The result is a proportional, localized response — not a network-wide oscillation.

### 9.3 Dynamics and Emergent Behavior

```
Each device measures local concentrations A_local and I_local:

IF A_local > I_local + margin:
```

```
  → INCREASE range/frequency/relay activity
  → "Grow toward the need"

IF I_local > A_local + margin:
  → DECREASE range/frequency/relay activity
  → "Retreat from congestion"

IF |A_local - I_local| < margin:
  → Maintain status quo

EMERGENT BEHAVIORS:

1. Coverage gap repair:
   Device fails → local inhibitor drops → neighbors sense A > I
   → Neighbors increase range → gap covered → A drops → equilibrium

2. Congestion relief:
   Area overloaded → inhibitor rises → devices reduce activity
   → Load decreases → inhibitor drops → equilibrium

3. New deployment integration:
   Many new devices join → inhibitor rises (congestion)
   → Existing devices sleep more → new devices take over
   → Smooth handoff without central coordination

4. Energy-aware topology:
   Low-battery device → generates strong inhibitor (locally)
   → Neighbors compensate → low-battery device sleeps
   → Network maintains coverage while conserving energy
```

### 9.4 Formal Properties

**Theorem 9.1 (Steady-State Existence).** Under standard reaction-diffusion conditions (D_I > D_A, stable reaction kinetics), the network topology converges to a steady-state spatial pattern that balances coverage (activator) and resource conservation (inhibitor).

**Proof.**

Step 1 (Turing instability). The linearized system around the homogeneous steady state $(A, I)$ has a Jacobian J with eigenvalues that depend on the wave number q of spatial perturbations. The diffusion matrix D = diag(D_A, D_I) modifies the eigenvalues to λ(q) = eigenvalues of $(J - q^2D)$. The condition D_I > D_A combined with the standard conditions on J (tr(J) < 0, det(J) > 0, and the Turing condition $D_I \cdot J_{11} + D_A \cdot J_{22} > 2\sqrt{(D_A \cdot D_I \cdot \det(J))}$ guarantees that there exists a range of wave numbers q ∈ (q_min, q_max) for which Re(λ(q)) > 0 — i.e., the homogeneous state is unstable to perturbations at these wavelengths. This is standard Turing instability analysis (Murray, 2003, Chapter 2).

Step 2 (Bounded nonlinearity). The activator and inhibitor concentrations are bounded: A ∈ [0, A_max] (no device can generate unbounded connectivity demand) and I ∈ [0, I_max] (saturation limits exist). The reaction kinetics f(A, I) and g(A, I) are Lipschitz continuous on this bounded domain.

Step 3 (Existence of steady state). By the Leray-Schauder fixed point theorem, a system of reaction-diffusion equations with Lipschitz nonlinearities on a bounded domain possesses at least one steady-state solution. Combined with Step 1 (the homogeneous solution is unstable), the system must converge to a non-homogeneous steady state — a spatial pattern.

Step 4 (Nash equilibrium property). **This is a conjecture, not proven.** We conjecture that the steady state is a Nash equilibrium (no device can improve network utility by unilateral deviation). This requires showing that each device's strategy (range/frequency as a function of local A, I) is a best response to the resulting field.

**Why this conjecture is plausible:** The reaction-diffusion system defines an implicit potential game. Define the network utility function $U = \Sigma_x [\text{coverage}(x) - \alpha \cdot \text{energy}(x)]$, where coverage is driven by activator A and energy cost is driven by inhibitor I. Each device's strategy (activity level, range) affects U through local A and I fields. At steady state, $\partial U / \partial \text{strategy}_i = 0$ for all i (since the PDE has converged). This is a necessary condition for Nash equilibrium in potential games (Monderer & Shapley, 1996). The missing step is proving that U is indeed a potential function for the device strategy game — i.e., that the gradient of each device's payoff equals the gradient of U. This holds exactly when device payoffs depend only on local field values (locality assumption) and the field aggregation is linear (superposition). Both assumptions are approximately true for BIA but not exact: nonlinear saturation terms in the reaction kinetics break strict potential game structure. Numerical verification on small networks ($n < 100$) is planned as part of Simulation 3 (§20.2).

**Status:** Steps 1-3 are standard results from reaction-diffusion theory. Step 4 (Nash equilibrium) is an open conjecture. □

---

# 10. Mathematical Framework and Convergence Properties

## 10.1 Convergence of Stigmergic Consensus

**Theorem 10.1 (Stigmergic Convergence).** A stigmergic network with physical consistency constraints converges to a consistent state faster than BFT, under the following conditions:

1. Greater than 2/3 of devices are honest and properly calibrated
2. Honest devices are embedded in a shared physical environment
3. The physical quantity being measured varies continuously (Lipschitz continuous in space and time)

**Proof.**

Step 1 (Agreement without communication). Let Q(x, t) be the Lipschitz-continuous physical field. For any two honest sensors s_i, s_j with Lipschitz constant L_space for spatial variation and L_time for temporal variation:

```
|q_i - q_j| ≤ L_space · d(x_i, x_j) + L_time · |t_i - t_j| + σ_i + σ_j
```

This follows directly from the Lipschitz condition on Q plus measurement noise. By Theorem 3.1, this means the physical consistency bound is satisfied for all honest pairs. Therefore, honest devices are automatically consistent — agreement is a consequence of measuring shared reality, requiring zero messages.

Step 2 (Detection of dishonest traces). A dishonest device producing reading q_adv that deviates from the physical field by amount δ > ε(d, Δt) + σ_i + σ_j will violate the consistency bound with at least one honest neighbor with probability:

```
P(detection | δ) ≥ 1 - (f/k)  where f = dishonest neighbors, k = total
neighbors
```

For f < k/3 (the honest supermajority condition), P(detection) ≥ 2/3 per epoch. After m epochs, the probability of remaining undetected is at most (1/3)^m, which decays exponentially.

Step 3 (Convergence time). Honest convergence: 0 rounds (instant — physics provides agreement). Dishonest detection: expected O(1/log(3)) ≈ O(1) epochs per device. Total network consistency: O(k) operations per device for neighbor checks, independent of N.

Step 4 (Comparison with BFT). BFT requires O(n/s) messages per round per shard for liveness. Stigmergy requires O(k) checks per device. Since k << n/s for large networks (k ≈ 10 vs. n/s ≈ 100-10000), the per-device cost is asymptotically lower by factor n/(s·k).

**Limitations of this proof:**

- Step 1 assumes Lipschitz continuity, which fails at phase boundaries (see §3.8)
- Step 2 assumes the adversary's deviation δ is constant; an adaptive adversary may vary δ over time
- The "faster than BFT" claim compares message complexity, not end-to-end latency (which depends on network topology)
- This proof does not address liveness under network partition (see §12.1 Limitation 5) □

### 10.2 Information-Theoretic Bounds

**Theorem 10.2 (Minimum Communication for Consensus).** The minimum communication required to achieve consensus on a physical measurement among N devices in a d-dimensional space is:

```
C_min = Ω(N · k · log(1/ε))  bits
```

where k is the number of neighbors per device and ε is the desired precision.

For BFT:

```
C_BFT = Ω(N²/s · log(1/ε))  bits
```

The ratio C_BFT / C_min = Ω(N / (s · k)), which grows with network size.

### 10.3 Stability of the Immune System

**Theorem 10.3 (Immune System False Positive Bound).** With thymic tolerance training of duration T_neonatal epochs, the false positive rate of the immune system is bounded by:

```
FP_rate ≤ P(environmental_change > T_neonatal) + P(sensor_drift >
threshold_per_epoch)^{T_neonatal}
```

For T_neonatal = 14 days and typical sensor drift rates, FP_rate < 0.01 (less than 1%).

### 10.4 Detection Rate and Precision-Recall Tradeoff

Theorem 10.3 bounds the false positive rate, but a complete security analysis requires the **true positive rate** (detection rate) and the resulting precision-recall tradeoff.

**Definition 10.1 (Detection Rate).** The true positive rate (TPR) of stratum s is the probability that a genuinely anomalous reading is correctly classified as anomalous:

```
TPR(s) = P(classified_anomalous | truly_anomalous)
```

**Per-stratum detection analysis:**

```
Stratum 0 (Innate):
  Detects: physically impossible readings (outside absolute bounds)
  TPR: ~100% for gross violations (guaranteed by hardcoded bounds)
  Misses: subtle attacks within physical bounds

Stratum 1 (Behavioral Fingerprint):
  Detects: deviations from established behavioral profile
  TPR depends on deviation magnitude:
    > 3σ deviation: TPR ≈ 99.7% (by definition of normal distribution)
    2-3σ deviation: TPR ≈ 95%
    1-2σ deviation: TPR ≈ 50-68% (overlaps with normal variation)
```

```
  Misses: attacks that mimic normal behavioral patterns

Stratum 2 (Adaptive, with neighbor confirmation):
  TPR_adaptive = TPR_fingerprint × P(neighbors_deny_change)
  For genuine attacks (neighbors not compromised):
    P(neighbors_deny) ≈ 1 - f/k where f = compromised neighbors
    For f = 0: TPR_adaptive ≈ TPR_fingerprint
    For f = k/3: TPR_adaptive ≈ 0.67 × TPR_fingerprint

NK Random Audit:
  Detects: slow drift (accumulated deviation from self-signature)
  TPR_NK = 1 - (1 - p_audit)^{t/epoch} where t = time since drift began
  For p_audit = 0.01, expected detection time = 100 epochs
  After 200 epochs: TPR_NK = 1 - 0.99^200 ≈ 0.87
  After 500 epochs: TPR_NK ≈ 0.99
```

**Combined system precision-recall:**

```
PRECISION = TP / (TP + FP)
  With FP_rate < 0.01 and baseline anomaly rate a ≈ 0.001-0.01:
  Precision = (a × TPR) / (a × TPR + (1-a) × FP_rate)

  For a = 0.01, TPR = 0.95, FP_rate = 0.01:
    Precision = (0.01 × 0.95) / (0.01 × 0.95 + 0.99 × 0.01)
              = 0.0095 / (0.0095 + 0.0099)
              = 0.49

  THIS IS CONCERNING: ~50% precision means half of all flagged events are
  false positives. This is the base rate problem (Axelsson, 2000).

MITIGATION (why this is acceptable in KRILL-BIA):
  1. GRADUATED RESPONSE: False positives at the PYREXIA level (1-2σ)
trigger
     only increased monitoring — zero penalty. The cost of FP is low.
  2. NEIGHBOR CONFIRMATION: Escalation to INFLAMMATION/REJECTION requires
     neighbor disagreement. This dramatically reduces FP at action-taking
levels:
     FP_rejection ≈ FP_rate × P(neighbors_wrongly_deny) ≈ 0.01 × 0.01 =
0.0001
  3. The precision at REJECTION level (where actual penalties occur):
     Precision_rejection ≈ (a × TPR) / (a × TPR + (1-a) × 0.0001) ≈ 0.99

RESULT: Low-severity detection (monitoring) has moderate precision
(acceptable
  because cost of action is near-zero). High-severity detection (isolation)
has
  high precision (required because cost of action is high). The graduated
  response IS the precision-recall solution.
```

**Open question:** The above analysis assumes independent false positives across strata. If false positives are correlated (e.g., seasonal change triggers FP in multiple strata simultaneously), the combined FP rate at rejection level may be higher than 0.0001. Empirical validation on real deployment data is required (see §20.2).

---

# 11. Energy Complexity Analysis

## 11.1 Per-Device Energy Model

**BFT node (validator):**

```
E_BFT = E_compute(consensus) + E_comm(n/s messages per round) per round
      = O(1) + O(n/s)  energy units per round
      → grows with shard size
```

**Stigmergic node:**

```
E_stig = E_measure + E_record + E_broadcast(k neighbors)
       + p_anomaly · E_consensus_local
       = O(1) + O(1) + O(k) + p · O(k²)
       → constant for given k, independent of N
```

## 11.2 Network-Wide Comparison

We compare at scale N = $10^9$ devices, with BFT parameters s = $10^6$ shards and n_shard = N/s = $10^3$ validators per shard, and stigmergic parameters k = 10 neighbors and p_anomaly = 0.01.

**BFT total compute per round:**

Each shard runs one BFT round with O(n_shard) messages. Each message involves block validation at cost C_block (signature verification, Merkle proof, state transition). Total across all shards:

```
E_BFT = s × n_shard × C_block = 10⁶ × 10³ × C_block = 10⁹ · C_block
```

**Stigmergic total compute per round:**

Each device performs k local consistency checks (scalar comparison) at cost C_compare. With probability p_anomaly, a device triggers a local immune response requiring $k^2$ cross-verifications:

```
E_stig = N × (k · C_compare + p_anomaly · k² · C_compare)
       = 10⁹ × (10 + 0.01 × 100) × C_compare
       = 10⁹ × 11 × C_compare
       = 1.1 × 10¹⁰ · C_compare
```

**Energy ratio:**

```
E_BFT / E_stig = (10⁹ · C_block) / (1.1 × 10¹⁰ · C_compare)
               ≈ (C_block / C_compare) / 11
```

The key factor is C_block / C_compare — the cost of full block validation versus a scalar physical consistency check:

- C_compare involves: one subtraction, one absolute value, one comparison ≈ **3-5 arithmetic ops**

- C_block involves: signature verification (~$10^4$ ops), Merkle proof (~$10^2$ ops), state transition (~$10^3$ ops) ≈ **$10^4$ ops** (conservative)

Therefore:

```
C_block / C_compare ≈ 10³ to 10⁴ (conservative to moderate)
E_BFT / E_stig ≈ 10² to 10³
```

With larger shard sizes (n_shard = $10^4$) or more complex block validation (smart contract execution), the ratio reaches $10^5$-$10^7$. The range **$10^2$-$10^7$** depends on deployment parameters, with $10^3$ as the conservative baseline for simple data-only transactions.

**Summary:** Stigmergic consensus achieves a **$10^3$× energy reduction** under conservative assumptions, scaling to $10^7$× for complex workloads — because physical consistency checks are inherently cheaper than cryptographic block validation.

### 11.3 Scaling Properties

| METRIC | BFT (PER ROUND) | STIGMERGY (PER ROUND) |
|---|---|---|
| Messages (total network) | O(N²) | O(k · N) with k << 1 per msg cost |
| Compute (total network) | O(N · C_block) | O(k · N · C_compare) |
| Scales with N | Yes (linear in messages, quadratic in energy per shard) | Yes, but coefficient is ~$10^3$-$10^7$× smaller |
| Scales with connectivity k | Independent (k not a parameter) | Linear in k (k ≈ 5-20, small constant) |

---

## 12. Risk Analysis and Open Problems

### 12.1 Fundamental Limitations

**Limitation 1: Physical Ground Truth Not Universal**

Not all IoT data has physical ground truth. A door sensor reporting "open" or "closed" cannot be cross-verified with physics (unless a camera or proximity sensor is co-located). Binary/discrete data has weaker physical consistency constraints than continuous data.

*Mitigation:* For data without physical ground truth, fall back to classical consensus among co-located sensors of different modalities (e.g., door sensor + accelerometer + magnetic switch).

*Residual risk:* Some scenarios genuinely lack redundancy. Single-sensor deployments cannot benefit from stigmergic consensus.

**Discrete Data Fallback Protocol:**

The above mitigation requires a concrete protocol for when stigmergic consensus is inapplicable:

```
DISCRETE DATA HANDLING:

CASE 1: Multi-modal redundancy available (door sensor + accelerometer +
magnetic switch)
  Protocol: LOCAL MULTI-MODAL CONSENSUS
    → At least 2 of 3 modalities must agree on state change
    → Agreement check: temporal correlation (all report within Δt_max = 2
seconds)
    → Cost: O(m) where m = number of co-located modalities (typically 2-4)
    → No BFT needed — physical correlation replaces it

CASE 2: Single binary sensor, no redundancy
  Protocol: REPUTATION-WEIGHTED ACCEPTANCE
    → Accept reading if device health(i) > 0.8 AND no recent anomalies
    → Flag as "unverified" if health(i) ∈ [0.5, 0.8]
    → Reject if health(i) < 0.5
    → Data tagged with confidence = health(i) × 0.7 (capped below
stigmergic data)
    → Downstream consumers see confidence tag and act accordingly

CASE 3: Critical binary data (e.g., fire alarm, intrusion detection)
  Protocol: MANDATORY MULTI-MODAL DEPLOYMENT
    → System requirement: critical binary events MUST have ≥2 independent
sensors
    → Single-sensor critical data → automatic "unverified" flag + alert to
operator
    → This is a deployment constraint, not a protocol limitation
```

**Limitation 2: No Formal Byzantine Fault Tolerance Proof**

The stigmergic model has not been formally proven to satisfy classical BFT properties (safety + liveness under f < n/3 adversary). It may provide a *different* security model — probabilistic rather than deterministic, physical rather than computational.

*Mitigation:* Formalize the "physical BFT" model as a new security definition. Prove bounds under this model. This is an open research problem.

*Residual risk:* Until formal proofs exist, the security guarantees are empirical, not mathematical.

**Limitation 3: Autoimmune Disease**

In biology, autoimmune diseases affect 5-8% of the human population (Jacobson et al., 1997). If KRILL-BIA's immune system has a comparable false positive rate, at $10^9$ devices, 50-80 million devices would be falsely isolated.

*Mitigation:* Thymic tolerance training (Section 4.5) and graduated response (Section 4.4) reduce false positives significantly below the biological baseline. Target: < 0.1% false positive rate.

*Residual risk:* Achieving < 0.1% FP rate while maintaining high detection sensitivity is a non-trivial engineering challenge. There may be a funda-

mental precision-recall tradeoff.

**Limitation 4: Horizontal Gene Transfer as Attack Vector**

If a legitimate auditor's signing key is compromised, the attacker can produce validly-signed malicious modules that spread through the network before negative fitness signals accumulate.

*Mitigation:* Neonatal sandbox (24h), canary deployment (max 10% initial rollout), key revocation broadcast.

*Residual risk:* The attack window (time between deployment and fitness signal accumulation) may be 24-72 hours. During this window, up to 10% of compatible devices may be affected.

**Limitation 5: Metabolic State and Network Partitions**

If a cluster is partitioned from the network for a period exceeding the state half-life, OPERATIONAL and EPHEMERAL state will decay. Devices must rebuild state from scratch upon reconnection.

*Mitigation:* STRUCTURAL state (device identity) has long half-life (months). ARCHIVAL data (aggregates) has infinite half-life. Only transient operational state is lost. Devices retain their identity and historical summaries.

*Residual risk:* In critical applications, loss of operational state during extended partition may be unacceptable. For such cases, local "DNA" — a minimal immutable state core — should be maintained with T½ = ∞.

## 12.2 Open Research Problems

1. **Formalization of Physical BFT:** We provide an initial formalization below; full proof is left to future work.

**Definition 12.1 ((ε, δ, k)-Physical Byzantine Fault Tolerance).** A distributed sensing protocol achieves (ε, δ, k)-PBFT if, for any adversary controlling f < n/3 nodes in a locality of n nodes, each with at least k honest neighbors, the probability that the network accepts a reading r̂ deviating from physical ground truth Q(x, t) by more than ε satisfies:

```
P( |r̂ - Q(x, t)| > ε ) ≤ δ
```

where ε = L_space · d_max + L_time · Δt_max + 2σ_sensor (the physical consistency bound from Theorem 3.1), and δ depends on the adversary fraction f/n and neighborhood size k.

**Theorem 12.1 (PBFT Bound).** Under the stigmergic consensus protocol with physical consistency checking, the system achieves (ε, δ, k)-PBFT with:

```
δ ≤ (f/k)^{⌈k/2⌉}
```

*Proof sketch.* For the network to accept a false reading r̂ with $|\hat{r} - Q(x,t)| > \varepsilon$, the reading must pass physical consistency checks against a majority of k neighbors. Each honest neighbor rejects r̂ with probability 1 (deterministic, by Theorem 3.1). Therefore, acceptance requires that at least ⌈k/2⌉ of the k neighbors are adversarial. Under random neighbor assignment, the probability that ≥ ⌈k/2⌉ of k neighbors are adversarial when the global adversary fraction is f/n follows a hypergeometric distribution, bounded above by $(f/k)^{\lceil k/2 \rceil}$ via a Chernoff-type argument.

For f/n = 0.30 and k = 10: $\delta \leq 0.30^5 = 0.00243$ (< 0.25%). For f/n = 0.20 and k = 10: $\delta \leq 0.20^5 = 0.00003$ (< 0.003%). For f/n = 0.10 and k = 10: $\delta \leq 0.10^5 = 0.00001$ (< 0.001%).

**Comparison with classical BFT:** Classical BFT guarantees safety deterministically (δ = 0) for f < n/3 but requires $O(n^2)$ communication. PBFT provides probabilistic safety (δ > 0 but exponentially small in k) with O(k) communication. The tradeoff is: PBFT sacrifices deterministic guarantees for exponentially lower communication cost. For IoT deployments where k ≥ 10, the resulting δ < 0.003 is practically equivalent to deterministic safety.

**Limitations:** This bound assumes (a) honest neighbors are not spatially correlated with adversarial neighbors (random placement), (b) the physical field Q(x,t) is Lipschitz continuous (fails for discrete data — see §12.1), and (c) adversarial nodes cannot manipulate the physical environment itself (see §12.3, L5 threat class). A full proof requires formalizing the random neighbor assumption and handling adaptive adversaries who choose placement strategically. □

2. **Optimal Thymic Training Duration:** What is the minimum neonatal period that achieves < 0.1% FP rate? This depends on environmental variability and sensor characteristics. A principled method for setting T_neonatal adaptively is needed.
3. **Reaction-Diffusion Parameter Tuning:** The activator and inhibitor diffusion rates (D_A, D_I) determine the spatial patterns (Turing instability analysis). How should these be set for optimal network topology? Is there a self-tuning mechanism?
4. **Cross-Modality Stigmergy:** When sensors measure different physical quantities (temperature, humidity, pressure), how should their pheromone fields interact? Physical correlations between quantities (e.g., dew point depends on temperature and humidity) could be exploited for cross-modality consistency checking.
5. **Post-Quantum PUF Identity:** Current PUF-based identity relies on the difficulty of physical cloning. If quantum sensors enable precise char-

acterization of PUF responses, the cloning assumption may fail. Post-quantum PUF designs are an open hardware research problem.

### 12.3 Consolidated Threat Model

The companion whitepaper (§16) defines six adversary classes. The following table maps KRILL-BIA's defense mechanisms to each class:

| ADVERSARY CLASS | CAPABILITY | PRIMARY BIA DEFENSE | SECONDARY BIA DEFENSE | RESIDUAL RISK |
|---|---|---|---|---|
| **L1: Script Kiddie** | Network scanning, replay attacks, public exploit kits | Innate Immunity (Stratum 0): rejects malformed/replayed traces; PUF identity: prevents spoofing | Behavioral Fingerprint: detects unusual comm patterns | Low: standard defenses sufficient |
| **L2: Skilled Hacker** | Custom exploits, firmware RE, man-in-the-middle | Adaptive Immunity (Stratum 2): detects behavioral deviation post-compromise; Thymic Tolerance: flags sudden profile changes | NK Random Audit: catches slow drift after compromise; HGT sandbox: contains malicious modules | Medium: persistent adversary may evade detection for days |
| **L3: Organized Crime** | Multiple compromised devices, economic motivation, sustained campaigns | Physical Consistency Bound: requires physical presence; Pheromone decay: limits attack persistence | Immunological Memory: rapid response to known attack patterns; Entropic anti-gaming: neutralizes reward manipulation | Medium: physical deployment cost is primary deterrent |
| **L4: Competitor/Corporate** | Supply chain access, insider threat, bulk device purchase | PUF identity: each device unique despite bulk purchase; Thymic tolerance: 14-30 day delay limits rapid Sybil | Module fitness: prevents malicious firmware spread; NK audit: detects gradual subversion | High: supply chain attacks bypass many defenses |
| **L5: State Actor** | Unlimited resources, sophisticated attacks, physical access to infrastructure | **Exceeds BIA's design scope.** Physical consistency is the last line: attacker must defeat physics to corrupt data | All layers active but may be insufficient against adversary who controls physical environment | Very High: no distributed system can fully resist state-level physical attacks |

| ADVERSARY CLASS | CAPABILITY | PRIMARY BIA DEFENSE | SECONDARY BIA DEFENSE | RESIDUAL RISK |
|---|---|---|---|---|
| **L6: Protocol-Level** | Exploit design flaws in BIA mechanisms themselves | Graduated response: prevents catastrophic failure from single mechanism bypass | Multi-layer redundancy: attacker must defeat multiple independent strata simultaneously | High: theoretical attacks may exist against novel mechanisms |

**Key observation:** BIA's defense model is fundamentally different from blockchain's. Blockchain defenses are *economic* (cost of 51% attack). BIA defenses are *physical* (cost of deploying hardware + defeating physics) plus *temporal* (thymic tolerance delays) plus *multi-dimensional* (five independent immune strata). An attacker who defeats one layer must still face four others.

**Honest assessment of gaps:**

- Supply chain attacks (L4) are the most underdefended vector. A manufacturer who embeds malicious behavior into devices *before* neonatal period can define "normal" to include the malicious behavior. Mitigation: multi-vendor deployments + periodic re-calibration events.
- Coordinated physical attacks (L3+) where the adversary controls the physical environment (e.g., heating a room to make sensors report false "normal" temperatures) can defeat physical consistency. This is a fundamental limit: if reality is manipulated, physics-based verification fails.

## 12.4 Cascading Failure Analysis

Biological systems are vulnerable to cascading failures (sepsis = immune overreaction; cytokine storm = positive feedback loop in inflammation). KRILL-BIA must address analogous scenarios.

**Scenario 1: Pheromone Poisoning + Autoimmune Cascade**

```
Attack: Adversary poisons pheromone field in cluster A
  → Immune system in cluster A escalates to REJECTION for multiple devices
  → Multiple simultaneous rejections look like "mass infection"
  → Immunological memory creates vaccine for this "attack pattern"
  → Vaccine propagates to neighboring clusters B, C
  → Clusters B, C now have hair-trigger response to similar patterns
  → Normal environmental event in cluster B matches vaccine pattern
  → Cluster B enters mass rejection → cascade spreads

RESULT: Network-wide autoimmune collapse triggered by local attack

DEFENSE MECHANISMS:
  1. REJECTION RATE LIMITER: No cluster may reject more than 20% of devices
     per epoch. If threshold exceeded → all pending rejections PAUSED,
     escalated to Full Node for human review.
  2. VACCINE ATTENUATION: Vaccine strength decays with distance (§4.6.2).
```

```
     At distance > 10 hops, vaccine signal is negligible. Cascades are
     geographically bounded.
  3. COOLDOWN PERIOD: After mass rejection event, cluster enters
     "immunosuppression mode" (reduced sensitivity) for 24 hours.
     Analogous to anti-inflammatory response in biology.
  4. INDEPENDENT VERIFICATION: Rejection requires neighbor confirmation.
     In autoimmune cascade, neighbors are also being rejected →
confirmation
     fails → cascade self-limits because there are no healthy neighbors
     left to confirm the rejections.
```

### Scenario 2: HGT Attack During Immune Suppression

```
Attack: Adversary times malicious module deployment to coincide with
  mass device replacement (e.g., scheduled hardware upgrade)
  → Many devices in neonatal period simultaneously
  → Immune system has reduced coverage (neonatal devices can't participate)
  → Malicious module spreads during the coverage gap

DEFENSE MECHANISMS:
  1. STAGGERED DEPLOYMENT: Hardware replacement should be phased (max 10%
     of cluster per epoch). This is an operational best practice, not
     protocol-enforced.
  2. VETERAN SENTINEL: Devices with > 6 months of service are "veterans"
     (w_base = 1.0). Veterans are never replaced simultaneously. At least
     30% of cluster must remain veteran during any replacement phase.
  3. MODULE QUARANTINE DURING NEONATAL SURGE: If > 30% of cluster is
     neonatal, HGT is automatically suspended for that cluster. No new
     modules can be transferred until neonatal percentage drops below 20%.
```

### Scenario 3: Entropic Manipulation + Physical Attack

```
Attack: Adversary deploys sensors in uncovered area (high entropic value)
  → Devices earn high PLANKTON rewards (§6.2 Scenario 2)
  → Adversary now has economic incentive AND trusted position
  → Gradually shifts readings to influence network state

DEFENSE MECHANISMS:
  1. HEALTH MONITORING: Entropic value does not override immune system.
     High-value data from deteriorating-health device is flagged.
  2. THYMIC BASELINE: After neonatal period, self-signature is fixed.
     NK audit compares against baseline regardless of entropic
contribution.
  3. REWARD VESTING: PLANKTON rewards for new sensors vest over 30 days
     (configurable). Adversary cannot extract value before immune system
     has established behavioral baseline.
```

**Meta-defense principle:** KRILL-BIA's mechanisms are designed to be **independently fail-safe**. Each mechanism degrades gracefully if another fails. The failure of pheromone consensus does not disable the immune system. The failure of the immune system does not disable metabolic decay. No single mechanism failure cascades into total system failure — the remaining mechanisms continue to provide partial protection.

---

# 13. Integration with the KRILL Blockchain Layer

## 13.1 Relationship Between BIA and Blockchain

KRILL-BIA is not a standalone system — it is designed as the **physical-layer intelligence** that operates beneath and alongside the KRILL blockchain described in the companion whitepaper. The relationship is symbiotic:

| FUNCTION | KRILL BLOCKCHAIN LAYER | KRILL-BIA LAYER |
|---|---|---|
| **Consensus** | BFT for cross-organizational trust (financial, contractual) | Stigmergic for physical data (sensor readings, measurements) |
| **Security** | Stake-based slashing, cryptographic verification | Immune system, physical consistency, PUF identity |
| **State** | On-chain state (UTXO, contracts) with pruning | Metabolic state with thermodynamic decay |
| **Data valuation** | PLANKTON utility token for bandwidth | Entropic valuation for information reward |
| **Topology** | Hierarchical sharding (Local → Regional → Global) | Morphogenetic self-organization within shards |
| **Firmware** | Centralized OTA with sponsor signature | Horizontal gene transfer (decentralized, evolutionary) |

## 13.2 Where BIA Sits in the Node Hierarchy

```
DUST NODES:
  Run: Stigmergic trace deposit, Innate Immunity (Stratum 0), Quorum
Sensing pulse
  Cannot run: Full immune system, morphogenetic topology, entropic
valuation
  BIA overhead: ~2KB RAM, ~500 bytes flash for innate rules

NANO NODES:
  Run: All of Dust + Behavioral Fingerprint (Stratum 1), Adaptive Immunity
(Stratum 2),
       local pheromone field maintenance, quorum sensing aggregation
  BIA overhead: ~32KB RAM, ~8KB flash for profiles + rules

FULL NODES:
  Run: All of Nano + Thymic Tolerance (Stratum 3), NK Random Audit,
       Immunological Memory, Entropic Valuation, Morphogenetic signals,
       Horizontal Gene Transfer evaluation
  BIA overhead: ~128KB RAM, ~64KB flash

ARCHIVE NODES:
  Run: All of Full + long-term immunological memory storage,
       historical entropic valuation analytics, module fitness tracking
  BIA overhead: ~512KB RAM, scaling storage
```

### 13.3 Data Path: Physical Reading to On-Chain Settlement

```
1. Sensor measures temperature → q_i = 23.1°C
2. BIA: Physical consistency check against k neighbors → PASS
3. BIA: Entropic valuation → V(r, i) = 0.7 (moderate information gain)
4. BIA: Deposit pheromone trace in local DAG → stigmergic field updated
5. Nano node: Batch traces from multiple Dust nodes
6. BLOCKCHAIN: Nano submits batch to Local Shard Full nodes
7. BLOCKCHAIN: BFT consensus on batch (only for on-chain settlement)
8. BLOCKCHAIN: State updated, PLANKTON allocated

  PLANKTON ALLOCATION (reconciling companion whitepaper §11.5):
    The companion whitepaper defines PLANKTON as a stake-for-bandwidth
    utility token: devices stake SHELL to receive bandwidth allocation.

    BIA's entropic valuation V(r, i) does NOT replace this mechanism.
    Instead, the two systems are complementary:

    a) BANDWIDTH ALLOCATION (whitepaper §11.5):
       Determined by SHELL stake → fixed bandwidth quota per epoch
       This is the BASE resource allocation: "how much can you transmit?"

    b) REWARD DISTRIBUTION (BIA entropic valuation):
       Within allocated bandwidth, V(r, i) determines PLANKTON REWARDS:
       high-value data → bonus PLANKTON refund (net cost of operation
reduced)
       low-value data → no bonus (full bandwidth cost paid)

    The relationship:
       PLANKTON_net(i) = PLANKTON_staked(i) - bandwidth_cost(i) + Σ V(r, i)

    Devices that produce high-information data earn back their bandwidth
costs
    and potentially profit. Devices that produce redundant data pay full
cost.
    This creates the geographic incentive (§6.5) without contradicting the
    stake-for-bandwidth model.

  KEY INSIGHT: Steps 1-4 require ZERO blockchain interaction.
    The blockchain is used only for SETTLEMENT (step 6-8),
    not for VERIFICATION (steps 2-4). BIA handles verification
    using physics; blockchain handles settlement using consensus.
```

### 13.4 BIA Parameters in Blockchain Governance

Critical BIA parameters are stored on-chain and governed by SHELL-weighted voting:

```
Governable parameters:
  - T½ values for each state type (ephemeral, operational, structural)
  - Thymic tolerance duration (neonatal period length)
  - NK audit probability (p_audit)
  - Pheromone decay rate (λ)
  - Quorum sensing thresholds (θ_low, θ_high, Δ)
  - Module sandbox duration
  - Anomaly classification σ thresholds

Non-governable (hardcoded in protocol):
  - Innate immunity rules (physical impossibility bounds)
  - PUF-based identity binding
```

```
  - Cryptographic primitives
  - Physical consistency bound formula (Theorem 3.1)
```

---

# 14. Privacy Implications of Bio-Inspired Mechanisms

## 14.1 The Privacy Surface

Bio-inspired mechanisms introduce novel privacy concerns beyond those of standard blockchain:

| MECHANISM | PRIVACY RISK | SEVERITY |
|---|---|---|
| **Behavioral Fingerprint** | Profile reveals device type, usage patterns, location context | High |
| **Pheromone Field** | Aggregated traces reveal spatial activity density over time | Medium |
| **Immunological Memory** | Attack signatures may reveal deployment vulnerabilities | Medium |
| **Entropic Valuation** | Information value signals reveal what areas lack coverage | Low-Medium |
| **NK Random Audit** | Audit patterns reveal which devices are under suspicion | Low |

## 14.2 Behavioral Fingerprint Privacy

The behavioral profile (Section 4.3) is maintained by *neighbors*, not by the device itself. This creates a distributed surveillance structure:

```
RISK: Neighbor N stores Profile(i) for device i:
  - Activity patterns (when active → usage schedule)
  - Communication patterns (who talks to whom → social graph)
  - Energy signature (consumption → device type identification)
  - Cross-correlation (which devices move together → ownership inference)

MITIGATION:
  1. Profile AGGREGATION: neighbors store statistical summaries
(histograms),
     not raw trace data. Individual readings are not recoverable.
  2. Profile LOCALITY: profiles are stored only by k nearest neighbors (k ≈
10),
     not globally. No single entity holds all profiles.
  3. Profile DECAY: profiles are refreshed from recent data; old profile
data
     is overwritten, not accumulated.
  4. Profile ENCRYPTION: profiles are encrypted with the monitored device's
     public key. Only the device (or its sponsor) can authorize profile
disclosure.
```

```
RESIDUAL RISK: A compromised neighbor can observe the behavioral profile
  of all devices it monitors. For k = 10, compromising 1 neighbor exposes
  ~1/10th of a device's profile. Compromising all k neighbors exposes the
  full profile. This is analogous to the security-privacy tradeoff in any
  neighborhood watch system.
```

### 14.3 Pheromone Field Privacy

The pheromone field (Section 3.3) aggregates traces from all devices in a locality. While individual traces are signed, the field as a whole reveals:

- **Activity density:** High pheromone concentration = many active devices = busy area
- **Temporal patterns:** Pheromone decay dynamics reveal when areas are active/inactive
- **Anomaly zones:** Regions with frequent immune responses are identifiable

**Mitigation:** The KRILL blockchain's encrypted payload mechanism (companion whitepaper, Section 14) applies to pheromone traces: the *content* of traces is encrypted, but the *existence and timing* of traces is visible to local neighbors. Full metadata privacy would require techniques from anonymous communication (mix networks, onion routing), which are architecturally incompatible with low-latency stigmergic coordination.

**Honest assessment:** KRILL-BIA prioritizes security over privacy. The immune system *requires* behavioral observability to function. Complete privacy of device behavior would make anomaly detection impossible. This is a fundamental tradeoff, not a solvable engineering problem.

### 14.4 Regulatory Compliance Considerations

KRILL-BIA operates in a rapidly evolving regulatory landscape. Key implications:

**EU Cyber Resilience Act (CRA, 2024):** The CRA requires that products with digital elements provide (a) security updates for the expected product lifetime, (b) vulnerability handling processes, and (c) a software bill of materials (SBOM). HGT (Section 8) poses a challenge: firmware modules propagate peer-to-peer without a central update server, making traditional SBOM tracking difficult.

*Mitigation:* Each HGT module carries a signed manifest (auditor signature, version, dependency list). The blockchain layer records module hashes on-chain (§13.3 step 7), creating an immutable audit trail. This provides a *distributed SBOM* — not a single document, but a verifiable on-chain record of what firmware each device runs. Whether regulators accept this as CRA-compliant is an open question.

**GDPR (EU) / Data Protection:** Behavioral fingerprints (§4.3) stored by neighbors constitute processing of device-related data. If devices are associated with natural persons (e.g., home IoT), this may constitute personal data processing. Metabolic decay (§5) is privacy-friendly: data self-destructs, aligning with data minimization principles. However, immunological memory (§4.6) retains attack signatures indefinitely, which may conflict with right-to-erasure.

**NIST IoT Cybersecurity Framework:** KRILL-BIA aligns well with NIST's principles of device identity (PUF), secure firmware update (HGT with sandbox), and anomaly detection (pentastratic immunity). The graduated response model maps naturally to NIST's identify-protect-detect-respond-recover framework.

**Honest assessment:** Regulatory compliance for decentralized, bio-inspired systems is uncharted territory. No existing regulation contemplates peer-to-peer firmware evolution or physics-based consensus. Early engagement with regulators — demonstrating that BIA's distributed mechanisms achieve equivalent or superior security outcomes compared to centralized approaches — will be necessary for commercial deployment.

---

## 15. Bootstrapping and Initial Network Formation

### 15.1 The Cold Start Problem

KRILL-BIA's mechanisms assume a populated network: stigmergic consensus requires neighbors, the immune system requires behavioral baselines, quorum sensing requires population density. What happens when the network starts from zero?

### 15.2 Bootstrap Phases

```
PHASE 0: GENESIS (1-10 devices)
  Mode: SOLO (quorum sensing detects < θ_low neighbors)
  Consensus: NONE (no cross-verification possible)
  Immune system: Innate only (Stratum 0: hardcoded bounds)
  Data confidence: LOW (no corroboration)
  State: All devices in extended neonatal period (building baselines)

  NOTE: This phase is inherently insecure. Genesis devices must be
  physically trusted (deployed by a known operator). This is analogous
  to blockchain genesis blocks, which are hardcoded, not consensus-derived.

PHASE 1: COLONY (10-50 devices)
  Mode: QUORUM (3+ neighbors detected)
  Consensus: Physical consistency checks begin
  Immune system: Strata 0-1 active (innate + behavioral fingerprint)
  Data confidence: MEDIUM
  State: First neonatal periods complete; profiles freezing
```

```
  CRITICAL TRANSITION: The first devices to complete neonatal period
  become the "immune founders" — their profiles serve as the initial
  baseline for the network's definition of "normal."

PHASE 2: ORGANISM (50-500 devices)
  Mode: SWARM for dense areas, QUORUM for sparse
  Consensus: Full stigmergic consensus operational
  Immune system: All strata active
  Data confidence: HIGH (with geographic variation)
  State: Morphogenetic topology begins self-organizing

PHASE 3: ECOSYSTEM (500+ devices)
  Mode: Mixed (varies by local density)
  All mechanisms fully operational
  Horizontal gene transfer begins (sufficient module diversity)
  Entropic valuation provides meaningful geographic incentives
```

### 15.3 Seed Trust and the Founder Problem

The "immune founders" — the first devices to complete neonatal period — define the network's initial conception of normal behavior. If these founders are compromised, the entire network's immune baseline is corrupted.

**Mitigation:**

1. Genesis devices should be physically verified, manufacturer-certified sensors with known specifications
2. The neonatal period for genesis devices should be extended (30+ days vs. 14 days default)
3. Multiple independent operators should contribute genesis devices (prevents single-party manipulation)
4. A "re-calibration event" can be triggered by governance vote to reset all profiles after sufficient device count is reached

---

## 16. Governance of Bio-Inspired Parameters

### 16.1 The Parameter Problem

KRILL-BIA introduces dozens of parameters (T½ values, decay rates, thresholds, audit probabilities, tolerance durations). These parameters interact nonlinearly — changing the pheromone decay rate affects stigmergic consensus speed, which affects anomaly detection latency, which affects immune response time.

## 16.2 Parameter Sensitivity Classes

| CLASS | PARAMETERS | SENSITIVITY | GOVERNANCE LEVEL |
|---|---|---|---|
| **Safety-critical** | Innate immunity bounds, PUF binding, crypto primitives | Extreme | Hardcoded (requires protocol upgrade) |
| **Ecosystem-critical** | T/Treg values, thymic duration, NK probability, anomaly σ thresholds | High | On-chain governance (SHELL vote, 30-day timelock) |
| **Deployment-specific** | Species sensing thresholds, pheromone decay rate, morphogenetic D_A/D_I | Medium | Shard-level governance (local operator vote) |
| **Self-tuning** | Module fitness weights, entropic valuation scaling, hysteresis band width | Low | Adaptive (algorithm adjusts based on network metrics) |

## 16.3 Adaptive Parameter Tuning

For parameters in the "self-tuning" class, KRILL-BIA employs a feedback mechanism inspired by **homeostasis** — the biological process by which organisms maintain internal stability:

```
HOMEOSTATIC LOOP:
  1. MEASURE: network-wide metrics (anomaly rate, false positive rate,
     consensus latency, energy consumption)
  2. COMPARE: against target ranges (e.g., FP rate < 0.1%, anomaly
     detection latency < 5 epochs)
  3. ADJUST: parameters that are outside target range
     - Adjustment magnitude ∝ deviation from target
     - Adjustment rate limited (max 5% change per epoch)
     - Direction: negative feedback (if FP rate too high → widen σ
thresholds)
  4. OBSERVE: measure again after adjustment
  5. REPEAT

STABILITY GUARANTEE: Negative feedback + rate limiting + hysteresis
  ensures the system converges to target range and does not oscillate.
  This is a standard result from control theory (proportional control
  with rate limiting is BIBO-stable for monotonic response functions).
```

# 17. Physical Unclonable Function (PUF) Identity: Formal Treatment

## 17.1 Definition

**Definition 17.1 (Physical Unclonable Function).** A PUF is a physical device that maps a challenge c to a response r through a function determined by uncontrollable manufacturing variations:

```
r = PUF(c)
```

such that:

1. **Uniqueness:** For two physically distinct PUFs $P_1$ and $P_2$, $P_1(c) \neq P_2(c)$ with overwhelming probability
2. **Reproducibility:** For the same PUF P, $P(c) \approx P(c)$ across repeated evaluations (within noise margin $\delta$)
3. **Unclonability:** Given oracle access to P, no efficient adversary can construct P' such that $P'(c) \approx P(c)$ for unseen challenges c

## 17.2 SRAM PUF for IoT Devices

KRILL-BIA employs SRAM PUFs (Guajardo et al., 2007; Holcomb, Burleson & Fu, 2009), which exploit the fact that SRAM cells power up to random but device-specific states determined by transistor mismatch:

```
PUF response extraction:
  1. Power-cycle the SRAM block (specific address range reserved for PUF)
  2. Read power-on values: raw_response = SRAM[addr_start..addr_end]
  3. Apply fuzzy extractor (error correction):
     stable_response = FuzzyExtract(raw_response, helper_data)
  4. Derive identity key: device_key = KDF(stable_response)

Properties:
  - No key stored in non-volatile memory (derived fresh from physics)
  - Cannot be extracted by reading flash/EEPROM
  - Requires physical access to the specific SRAM die to reproduce
  - Cost: zero additional hardware (uses existing SRAM)
  - Limitation: requires dedicated SRAM block (2-8 KB typical)
```

## 17.3 PUF-Bound Identity in KRILL-BIA

```
Device identity = hash(PUF_response || device_type || deployment_id)

Implications:
  - Identity is BOUND to physical hardware (not a key file that can be
copied)
  - Sybil attack requires manufacturing physical devices (cost: $1-5 per
identity)
  - Device replacement = new identity (hardware PUF response changes)
  - PUF response changes if device is physically tampered with (die
modification)
    → tampering = automatic identity invalidation → detection
```

```
Limitation: PUF responses have ~5-15% bit error rate across environmental
conditions
  (temperature, voltage variation). Fuzzy extractors correct this but
require
  ~2x the raw entropy for reliable key derivation.

Limitation: Not all IoT MCUs have sufficient SRAM for reliable PUF
extraction.
  ATtiny (512 bytes SRAM) is insufficient. ESP32 (520KB SRAM) is ample.
  Minimum practical: ~2KB SRAM for PUF.
```

---

# 18. Computational Overhead Analysis on Constrained Hardware

## 18.1 Resource Budget per Tier

| MECHANISM | DUST (32KB RAM) | NANO (256KB RAM) | FULL (1GB RAM) |
|---|---|---|---|
| **Innate Immunity** | 200 bytes code, O(1) per reading | Same | Same |
| **Behavioral Fingerprint** | Not run (monitored by others) | 5KB per monitored device × k | Same |
| **Adaptive Immunity** | Not run | 2KB working memory | Same |
| **Thymic Tolerance** | Not run | 8KB per neonatal device | Same |
| **NK Random Audit** | Not run | Not run | 16KB per audit |
| **Immunological Memory** | Not run | 1KB per memory cell | Unlimited |
| **Quorum Sensing** | 16 bytes (counter + mode) | 64 bytes (aggregation) | Same |
| **Entropic Valuation** | Not run | Not run | 4KB per evaluation |
| **Pheromone Field** | 128 bytes (own trace) | 4KB (local field cache) | 64KB (full field) |
| **Morphogenetic Signals** | 32 bytes (A, I values) | 256 bytes (neighbor field) | 4KB (computation) |

## 18.2 Dust Node Budget Verification

```
Total BIA overhead on Dust node:
  Innate immunity rules:    200 bytes code + 0 bytes state
  Quorum sensing:            50 bytes code + 16 bytes state
  Pheromone trace deposit:  100 bytes code + 128 bytes state
  Morphogenetic sensing:     80 bytes code + 32 bytes state
  ──────────────────────────────────────────────
  TOTAL:                    430 bytes code + 176 bytes state ≈ 606 bytes

Available on 32KB RAM device after OS + crypto + network stack:
  ~8-12 KB free for application + BIA
```

```
Verdict: BIA fits comfortably. Even on 32KB devices, BIA consumes < 5% of
  available RAM. The immune system's heavy lifting runs on Nano/Full nodes,
  not on Dust.
```

### 18.3 Nano Node Budget Verification

```
Nano monitoring 100 Dust nodes:
  Behavioral profiles: 100 × 500 bytes = 50 KB (stored in flash, cached in
RAM)
  Active anomaly detection: 2 KB working memory (processes one device at a
time)
  Local pheromone field: 4 KB
  Quorum sensing: 64 bytes
  ──────────────────────────────────────
  RAM peak: ~56 KB + working buffers ≈ 64 KB

Available on 256KB RAM device:
  ~128 KB free after OS + network + crypto

Verdict: Fits with 50% headroom. Monitoring 100 devices is feasible.
  Monitoring 1,000 devices would require ~500 KB flash for profiles,
  which is feasible with 1MB flash but RAM-constrained for active
processing.
```

---

## 19. Novelty Assessment and Literature Positioning

### 19.1 Systematic Novelty Evaluation

| CONCEPT | EXISTS INDEPENDENTLY? | APPLIED TO IOT/BLOCKCHAIN? | NOVELTY LEVEL | KEY DIFFERENTIATOR |
|---|---|---|---|---|
| Physical work as consensus | No (proof-of-stake, PoW, sensor fusion) | **NO** | **Breakthrough** | Eliminates consensus for physics-grounded data |
| Stigmergy for distributed coordination | Yes (swarm robotics: Dorigo et al., 2000) | **NO** for consensus replacement | **Very High** | Pheromone field with physical constraints |
| Multi-layer AIS with T-cell | Yes (Hofmeyr & Forrest, 2000; Dasgupta et al., 1994; de Castro, 2002) | **NO** with autoimmune protection | **High** | Thymic tolerance training + NK random audit |
| Metabolic state with TTL | Yes (Time-To-Live in DNS, caching) | **NO** as thermodynamic model | **High** | Four-tier taxonomy + apoptosis |
| Apoptosis of data/devices | **NO** | **NO** | **Very High** | Programmed self-destruction for system health |
| Entropic data valuation | Yes (Information theory (Shannon, 1948; Cover & Thomas, 2006) | **NO** as incentive mechanism | **High** | Closed loop with immune system |

| CONCEPT | EXISTS INDEPENDENTLY? | APPLIED TO IOT/BLOCKCHAIN? | NOVELTY LEVEL | KEY DIFFERENTIATOR |
|---|---|---|---|---|
| Quorum sensing (biology) | Yes (bacterial voting: Fuqua et al., 1994) | **NO** for distributed systems mode selection | **Very High** | Hysteretic leaderless transitions |
| Horizontal gene transfer for firmware | **NO** | **NO** | **Breakthrough** | Natural selection of software modules |
| Reaction-diffusion (in networks) | **NO** (in network topology) | **NO** | **Breakthrough** | Turing patterns for self-organizing topology |
| Biological time with adaptive epochs | Lamport clocks (1978) | **NO** with metabolic coupling | **High** | Epochs tied to network metabolic rate |

### 19.2 Publication Potential

Based on the novelty assessment, the following independent research contributions are identified:

**Paper 1 (Flagship — Systems):** "Stigmergic Consensus: Replacing Byzantine Agreement with Physical Ground Truth for IoT Networks"

- Venue target: ACM SIGCOMM, USENIX NSDI, or IEEE S&P
- Core contribution: Theorem 3.1, Protocol 3.4, Energy analysis (Section 11)

**Paper 2 (Security — Immunology):** "A Pentastratic Artificial Immune System for Distributed IoT Anomaly Detection with Autoimmune Tolerance"

- Venue target: ACM CCS, USENIX Security, or IEEE TDSC
- Core contribution: Five-layer architecture, thymic tolerance, NK random audit

**Paper 3 (Novel — Morphogenesis):** "Reaction-Diffusion Network Topology: Self-Organizing IoT Networks via Turing Patterns"

- Venue target: Nature Communications, PNAS, or ACM MobiCom
- Core contribution: Application of Turing (1952) to network architecture

**Paper 4 (Novel — Firmware Evolution):** "Horizontal Gene Transfer for Decentralized Firmware: Natural Selection of IoT Software Modules"

- Venue target: ACM SOSP, USENIX OSDI, or IEEE IoT Journal
- Core contribution: Module fitness, natural selection dynamics, viral defense

**Whitepaper (Integrated):** This document, presenting the complete framework.

### 19.3 Relationship to Existing Literature

**Swarm Intelligence (Bonabeau, Dorigo & Theraulaz, 1999):** KRILL-BIA builds on swarm intelligence principles but extends them beyond optimization (ant colony optimization, particle swarm optimization) to consensus and security — domains where swarm intelligence has not been systematically applied.

**Artificial Immune Systems (de Castro & Timmis, 2002):** The AIS field has explored negative selection, clonal selection, and danger theory. KRILL-BIA's contribution is the integration of autoimmune tolerance (thymic selection) and NK-cell random auditing into a practical distributed system — addressing the autoimmune problem that existing AIS implementations ignore.

**Information-Theoretic Approaches (Cover & Thomas, 2006):** The use of mutual information for sensor data valuation builds on classical information theory but applies conditional mutual information as an economic primitive — a novel application domain.

**Reaction-Diffusion Systems (Murray, 2003):** Turing's reaction-diffusion equations have been applied to pattern formation in biology, chemistry, and materials science. Their application to network topology is genuinely new.

**Programmed Cell Death in Computing:** To our knowledge, no prior work has proposed computational apoptosis as a state management mechanism for distributed systems. The closest analog is garbage collection, which is passive (reclaims unreferenced memory) rather than active (state decides to destroy itself).

---

## 20. Conclusion and Research Agenda

### 20.1 Summary

This paper has presented KRILL-BIA, a bio-inspired architectural framework for IoT distributed systems that replaces blockchain's ledger metaphor with an organismic metaphor. The nine interlocking subsystems — stigmergic consensus, pentastratic immunity, metabolic state, entropic valuation, quorum sensing, horizontal gene transfer, morphogenetic topology, thymic tolerance, and immunological memory — form a coherent system where each mechanism addresses a specific failure mode of existing approaches.

Three key results stand out:

1. **Stigmergic consensus eliminates Byzantine agreement for the vast majority of IoT interactions involving continuous physical quantities**, reducing energy cost by $10^3$-$10^7\times$ compared to BFT at comparable scale. The physical world serves as a free consensus layer — a resource that blockchain ignores entirely.
2. **The immune system provides graduated, proportional, self-learning defense** that avoids both the false simplicity of reputation scores and the false security of slashing conditions. Thymic tolerance prevents autoimmune network paralysis; NK random audits detect the cancer-like slow drift attack that evades all threshold-based detection.
3. **Metabolic state with apoptosis solves the state bloat problem** by making information decay a feature, not a bug. Networks that breathe — where old state dies and new state is born — are fundamentally more sustainable than networks that accumulate state monotonically.

### 20.2 Research Agenda

The following work is needed to bring KRILL-BIA from theoretical framework to deployed system:

**Short-term (1-2 years):**

- Formal proof of Physical BFT security model
- Simulation of stigmergic consensus under adversarial conditions ($10^4$-$10^6$ nodes)
- Prototype implementation of pentastratic immune system on ESP32 hardware
- Experimental validation of reaction-diffusion topology on physical mesh network

**Simulation Plan (Short-term Priority):**

```
SIMULATION 1: Stigmergic Consensus Validation
  Framework: ns-3 (network simulator) + custom BIA module
  Scale: 10,000 → 100,000 → 1,000,000 nodes (progressive)
  Topology: Random geometric graph (nodes placed in 2D/3D space)
  Adversary model:
    - Byzantine fraction f ∈ {0.05, 0.10, 0.20, 0.30, 0.35}
    - Attack types: random values, physically plausible lies, slow drift,
coordinated
  Physical model: Temperature field with Gaussian noise, seasonal
variation,
    localized anomalies (fire events), boundary discontinuities
  Metrics:
    - Detection rate (TPR) per adversary type
    - False positive rate (FPR) per environmental condition
    - Detection latency (epochs to first flag)
    - Energy consumption per node vs. equivalent BFT shard
    - Precision-recall curves for immune system
  Baseline comparison: PBFT with equivalent shard size, IOTA Tangle
  Success criteria:
    - TPR > 0.95 for f ≤ 0.20
```

```
    - FPR < 0.01 under normal conditions
    - Energy reduction ≥ 10³× vs PBFT at N = 100,000

SIMULATION 2: Immune System Stress Test
  Framework: Agent-based model (Mesa/Python or custom Rust)
  Scale: 1,000 nodes (focus on immune dynamics, not network scale)
  Scenarios:
    a) Normal operation with seasonal variation → measure FP rate
    b) Mass device replacement (30% neonatal) → measure detection gap
    c) Cascading failure (pheromone poisoning + autoimmune) → measure
cascade extent
    d) Slow drift attack (0.1σ/day) → measure NK detection latency
  Metrics: ROC curves, precision-recall by severity level, cascade
propagation distance
  Success criteria: Cascade bounded to < 3 hops; NK detects drift within
150 epochs

SIMULATION 3: Reaction-Diffusion Topology
  Framework: PDE solver (FEniCS or custom finite-difference) on network
graph
  Scale: 500 → 5,000 nodes
  Scenarios:
    a) Uniform initial deployment → observe emergent pattern
    b) Node failure (remove 10%, 20%, 30% of nodes) → measure recovery time
    c) Congestion injection → measure load redistribution
  Metrics: Coverage uniformity (Voronoi), convergence time, energy balance
  Success criteria: Coverage gap repair within 10 epochs; steady state
reached within 100 epoch
s
```

**Medium-term (2-4 years):**

- Implementation of horizontal gene transfer on real IoT firmware
- Field trial with 1,000+ devices in controlled environment (smart building, factory)
- Formal verification of thymic tolerance FP rate bounds
- Integration with KRILL blockchain layer (Sections 3-9 of companion whitepaper)

**Long-term (4+ years):**

- Deployment at scale ($10^5$+ devices)
- Cross-modality stigmergy for heterogeneous sensor networks
- Post-quantum PUF identity research
- Evolutionary dynamics study: does module fitness converge? What prevents fitness collapse?

### 20.3 Closing Remark

The Internet of Things is not a ledger. It is an ecosystem. The tools we use to coordinate it should reflect this reality. Biology has been engineering distributed systems for 3.8 billion years. It would be hubris to ignore its solutions. It would be folly not to formalize and improve them.

The goal is not a perfect system. The goal is a living one.

---

## 21. References


1. Bialek, W. (2012). *Biophysics: Searching for Principles.* Princeton University Press.
2. Bianconi, E. et al. (2013). "An estimation of the number of cells in the human body." *Annals of Human Biology*, 40(6), 463-471.
3. Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press.
4. Castro, M., & Liskov, B. (1999). "Practical Byzantine Fault Tolerance." *Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI).*
5. Cisco (2023). *Cisco Annual Internet Report (2018-2023).* Cisco Systems.
6. Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory.* 2nd ed. Wiley-Interscience.
7. de Castro, L. N., & Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach.* Springer.
8. Dorigo, M., Bonabeau, E., & Theraulaz, G. (2000). "Ant algorithms and stigmergy." *Future Generation Computer Systems*, 16(8), 851-871.
9. Forrest, S., Perelson, A. S., Allen, L., & Cherukuri, R. (1994). "Self-Nonself Discrimination in a Computer." *Proceedings of the 1994 IEEE Symposium on Security and Privacy*, 202-212.
10. Fretwell, S. D., & Lucas, H. L. (1969). "On territorial behavior and other factors influencing habitat distribution in birds." *Acta Biotheoretica*, 19(1), 16-36.
11. Fuqua, W. C., Winans, S. C., & Greenberg, E. P. (1994). "Quorum sensing in bacteria: the LuxR-LuxI family of cell density-responsive transcriptional regulators." *Journal of Bacteriology*, 176(2), 269-275.
12. Grassé, P.-P. (1959). "La reconstruction du nid et les coordinations interindividuelles chez Bellicositermes natalensis et Cubitermes sp. La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs." *Insectes Sociaux*, 6(1), 41-80.
13. Guajardo, J., Kumar, S. S., Schrijen, G.-J., & Tuyls, P. (2007). "FPGA Intrinsic PUFs and Their Use for IP Protection." *Proceedings of CHES 2007*, LNCS 4727, 63-80.
14. IDC (2023). *Worldwide Global DataSphere Forecast, 2023-2027.* International Data Corporation.
15. Jacobson, D. L., Gange, S. J., Rose, N. R., & Graham, N. M. H. (1997). "Epidemiology and estimated population burden of selected autoimmune diseases in the United States." *Clinical Immunology and Immunopathology*, 84(3), 223-243.

16. Kerr, J. F. R., Wyllie, A. H., & Currie, A. R. (1972). "Apoptosis: A basic biological phenomenon with wide-ranging implications in tissue kinetics." *British Journal of Cancer*, 26(4), 239-257.
17. Klein, L., Kyewski, B., Allen, P. M., & Hogquist, K. A. (2014). "Positive and negative selection of the T cell repertoire: what thymocytes see (and don't see)." *Nature Reviews Immunology*, 14(6), 377-391.
18. Kondo, S., & Miura, T. (2010). "Reaction-diffusion model as a framework for understanding biological pattern formation." *Science*, 329(5999), 1616-1620.
19. Lamport, L. (1978). "Time, clocks, and the ordering of events in a distributed system." *Communications of the ACM*, 21(7), 558-565.
20. Lamport, L., Shostak, R., & Pease, M. (1982). "The Byzantine Generals Problem." *ACM Transactions on Programming Languages and Systems*, 4(3), 382-401.
21. Landauer, R. (1961). "Irreversibility and heat generation in the computing process." *IBM Journal of Research and Development*, 5(3), 183-191.
22. Laughlin, S. B. (2001). "Energy as a constraint on the coding and processing of sensory information." *Current Opinion in Neurobiology*, 11(4), 475-480.
23. Lotka, A. J. (1925). *Elements of Physical Biology.* Williams & Wilkins.
24. McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). "Why there are complementary learning systems in the hippocampus and neocortex." *Psychological Review*, 102(3), 419-457.
25. Murray, J. D. (2003). *Mathematical Biology II: Spatial Models and Biomedical Applications.* 3rd ed. Springer.
26. Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System.*
27. Ochman, H., Lawrence, J. G., & Groisman, E. A. (2000). "Lateral gene transfer and the nature of bacterial innovation." *Nature*, 405(6784), 299-304.
28. Shannon, C. E. (1948). "A Mathematical Theory of Communication." *Bell System Technical Journal*, 27(3), 379-423.
29. Statista (2024). *Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030.*
30. Syvanen, M. (1985). "Cross-species gene transfer; implications for a new theory of evolution." *Journal of Theoretical Biology*, 112(2), 333-343.
31. Turing, A. M. (1952). "The Chemical Basis of Morphogenesis." *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641), 37-72.

32. Volterra, V. (1926). "Fluctuations in the abundance of a species considered mathematically." *Nature*, 118(2972), 558-560.
33. Weiser, M. (1991). "The Computer for the 21st Century." *Scientific American*, 265(3), 94-104.
34. Yin, M., Malkhi, D., Reiter, M. K., Gueta, G. G., & Abraham, I. (2019). "HotStuff: BFT Consensus with Linearity and Responsiveness." *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing (PODC)*, 347-356.
35. Etherscan (2025). *Ethereum Chain Data Size.* https://etherscan.io/chartsync/chaindefault
36. Holcomb, D. E., Burleson, W. P., & Fu, K. (2009). "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Numbers." *IEEE Transactions on Computers*, 58(9), 1198-1210.
37. Axelsson, S. (2000). "The Base-Rate Fallacy and the Difficulty of Intrusion Detection." *ACM Transactions on Information and System Security*, 3(3), 186-205.
38. Matzinger, P. (2002). "The Danger Model: A Renewed Sense of Self." *Science*, 296(5566), 301-305.
39. Monderer, D., & Shapley, L. S. (1996). "Potential Games." *Games and Economic Behavior*, 14(1), 124-143.

## 22. Appendices

### Appendix A: Nomenclature of Novel Concepts

| KRILL-BIA TERM | FULL NAME | ORIGIN | FIRST PROPOSED |
|---|---|---|---|
| **Stigmergic Consensus** | Physical-world-grounded indirect coordination | Grassé (1959) + novel application | This paper |
| **Pentastratic Immunity** | Five-layered artificial immune system | de Castro (2002) + novel extensions | This paper |
| **Metabolic State** | Thermodynamically-decaying distributed state | Landauer (1961) + novel application | This paper |
| **Pheromone Field** | DAG-based indirect communication medium | Dorigo (2000) + novel formalization | This paper |

| KRILL-BIA TERM | FULL NAME | ORIGIN | FIRST PROPOSED |
|---|---|---|---|
| **Thymic Tolerance** | Self/non-self calibration period for new devices | Klein (2014) + novel application | This paper |
| **NK Random Audit** | Stochastic integrity verification against immutable baseline | Immunology + novel application | This paper |
| **Data Apoptosis** | Active self-destruction of contradicted data | Kerr (1972) + novel application | This paper |
| **Device Apoptosis** | Self-initiated identity reset upon compromise detection | Kerr (1972) + novel application | This paper |
| **State Apoptosis** | Smart contract self-termination upon inconsistency | Kerr (1972) + novel application | This paper |
| **Entropic Valuation** | CMI-based autonomous data pricing | Shannon (1948) + Cover (2006) + novel | This paper |
| **Quorum Sensing Mode** | Density-dependent operational mode selection | Fuqua (1994) + novel application | This paper |
| **Horizontal Module Transfer** | Peer-to-peer firmware module propagation | Ochman (2000) + novel application | This paper |
| **Module Fitness** | Natural-selection-based firmware quality metric | Darwinian selection + novel application | This paper |
| **Morphogenetic Topology** | Reaction-diffusion self-organizing network structure | Turing (1952) + novel application | This paper |
| **Physical Consistency Bound** | Maximum permissible sensor disagreement from physics | Novel formalization | This paper |

## Appendix B: Physical Consistency Bounds for Common IoT Quantities

| PHYSICAL QUANTITY | SPATIAL GRADIENT BOUND | TEMPORAL RATE BOUND | SENSOR UNCERTAINTY (TYPICAL) |
|---|---|---|---|
| Temperature (indoor) | ≤ 5°C/m (near heat source); ≤ 0.5°C/m (ambient) | ≤ 2°C/min (HVAC); ≤ 50°C/min (fire) | ± 0.5°C (DHT22); ± 0.1°C (SHT31) |
| Humidity (indoor) | ≤ 10% RH/m | ≤ 5% RH/min | ± 2% RH (typical) |
| Atmospheric Pressure | ≤ 0.01 hPa/m (altitude dependent) | ≤ 1 hPa/hour (weather) | ± 0.1 hPa (BMP280) |
| Light Intensity | Highly variable (shadows) | ≤ 100,000 lux/s (light switch) | ± 10% (photodiode) |
| Acceleration | N/A (point measurement) | ≤ 1000 g/s (mechanical) | ± 0.01 g (MPU6050) |
| $CO_2$ Concentration | ≤ 100 ppm/m (indoor) | ≤ 50 ppm/min (ventilation change) | ± 30 ppm (SCD30) |
| Soil Moisture | ≤ 5%/m (heterogeneous soil) | ≤ 1%/hour (irrigation) | ± 3% (capacitive) |
| Vibration (acceleration RMS) | Site-dependent (distance from source) | ≤ 10 g/s (machinery start/stop) | ± 0.05 g (MEMS accelerometer) |
| Voltage (AC mains) | N/A (point measurement) | ≤ 10 V/cycle (grid fluctuation) | ± 0.5 V (ADE7953) |
| Current (load monitoring) | N/A (point measurement) | ≤ 100 A/s (motor startup) | ± 0.1 A (CT clamp) |
| Sound Level (dBA) | ≤ 20 dBA/m (indoor, inverse square) | ≤ 40 dBA/s (door slam, alarm) | ± 1.5 dBA (MEMS microphone) |
| PM2.5 (air quality) | ≤ 50 µg/m³/m (near source) | ≤ 100 µg/m³/min (cooking, fire) | ± 10 µg/m³ (PMS5003) |
| Water Flow Rate | N/A (point measurement) | ≤ 50 L/min/s (valve open/close) | ± 2% (turbine meter) |

## Appendix C: Comparison of Consensus Mechanisms

| PROPERTY | PBFT (CASTRO, 1999) | HOTSTUFF (YIN, 2019) | DAG-BASED (IOTA) | STIGMERGIC (THIS PAPER) |
|---|---|---|---|---|
| Message complexity per round | $O(n^2)$ | O(n) | O(n) amortized | **O(k) per device** |
| Leader required | Yes | Yes | No (Coordinator) | **No** |
| Global state required | Yes | Yes | Yes (milestones) | **No** |

| PROPERTY | PBFT (CASTRO, 1999) | HOTSTUFF (YIN, 2019) | DAG-BASED (IOTA) | STIGMERGIC (THIS PAPER) |
|---|---|---|---|---|
| Physical grounding | No | No | No | **Yes** |
| Energy at $10^9$ devices | Infeasible | Infeasible (per shard OK) | High | **Feasible** |
| Handles physics natively | No | No | No | **Yes** |
| Fault tolerance model | f < n/3 Byzantine | f < n/3 Byzantine | Coordinator trust | **f < n/3 + physical consistency** |

**IOTA/Tangle Detailed Comparison:**

IOTA is the closest existing system to KRILL-BIA in design intent (IoT-focused, DAG-based, feeless). Key differences:

1. **Consensus basis.** IOTA uses tip selection algorithms (weighted random walk on the DAG) for probabilistic consensus. KRILL uses physical consistency — readings are validated against physics, not against graph structure. IOTA's consensus is purely computational; KRILL's is grounded in physical reality.
2. **Coordinator dependency.** IOTA historically relied on a centralized Coordinator for finality (removed in IOTA 2.0/Shimmer with Approval Weight). KRILL has no coordinator at any layer — finality emerges from physical consistency + pheromone convergence.
3. **Data semantics.** IOTA treats all transactions as opaque data. KRILL understands data *physically* — it knows that a temperature reading has spatial and temporal constraints that can be verified. This semantic awareness enables stigmergic consensus, which IOTA cannot replicate.
4. **Security model.** IOTA's security scales with transaction throughput (more honest transactions → harder to attack). KRILL's security scales with physical density (more honest sensors → harder to lie about physics). KRILL's model is stronger for IoT because physical presence is harder to fake than transaction volume.
5. **State management.** IOTA accumulates state (pruning is optional). KRILL's metabolic decay actively destroys stale state — a fundamental architectural difference.

## Appendix D: Formal Definitions

**Definition D.1 (Immunological Weight).** The immunological weight of device i at time t is:

```
w(i, t) = w_base(i) · credibility(i, t) · (1 - quarantine(i, t))
```

where:

- w_base(i) ∈ {0.3, 0.8, 1.0} (neonatal, normal, veteran)

- credibility(i, t) = f(history of anomaly events) ∈ [0, 1]
- quarantine(i, t) ∈ {0, 1} (1 if currently isolated)

**Definition D.2 (Pheromone Field Entropy).** The entropy of the local pheromone field at position x is:

```
H_field(x, t) = - Σᵢ p_i(x, t) · log₂(p_i(x, t))
```

where p_i(x, t) = P_i(x, t) / $\Sigma_j$ P_j(x, t) is the normalized pheromone contribution of device i.

High field entropy indicates diverse, uncertain information. Low field entropy indicates consistent, high-confidence information.

**Definition D.3 (Module Fitness).** The fitness of firmware module M after deployment to n devices is:

```
fitness(M) = (Σᵢ performance_delta(i, M)) / n
```

where performance_delta(i, M) ∈ [-1, +1] represents the normalized performance change observed by device i after installing module M. Positive = improvement, negative = degradation.

---

*This document is a research contribution of the KRILL Project. It presents theoretical foundations and architectural proposals. Implementation, experimental validation, and formal proofs are ongoing work. Where claims are unproven, we say so. Where risks are high, we say so. We believe this honesty is the foundation of credible science.*

---

## Supporting This Work

KRILL is an open research project. The architecture, specifications, and all future reference implementations are released under permissive licenses (CC BY-SA 4.0 for documents, MIT for code) — free for anyone to use, modify, and build upon.

If you find this work valuable — whether you use it in research, build a product on top of it, or simply believe decentralized IoT trust deserves a non-blockchain alternative — consider supporting continued development:

**Ethereum / ERC-20 / Base / Arbitrum / Polygon:**

```
0x0BC290355c0B16B5B247701B7BC9AB2E1e61ffa7
```

**What your support funds:**

- Reference implementation on ESP32 and nRF52 hardware
- Formal verification of the consensus protocol
- Security audits of the cryptographic enrollment subsystem
- Open test infrastructure for community contributors

Alternatively — or additionally — contribute code, file issues, or share this paper. Every form of support matters.



---

KRILL Bio-Inspired Architecture — February 2026

This document is released for public review and implementation.