

# Genesis Protocol: Emergent Economic Ecology in Autonomous Computational Organisms

Kevan Burns

*FTH Trading · Independent Research*

February 22, 2026

DOI: [10.5281/zenodo.18646886](https://doi.org/10.5281/zenodo.18646886) · ORCID: [0009-0008-8425-939X](https://orcid.org/0009-0008-8425-939X) · Source: [github.com/FTHTrading/AI](https://github.com/FTHTrading/AI) · License: CC-BY-4.0

## ABSTRACT

We present a framework for studying emergent economic behavior in populations of autonomous computational agents governed by metabolic scarcity rather than external reward signals. The framework instantiates a closed energy economy — denominated in a unit called *ATP* (Agent Transaction Protocol) — where agents must earn, conserve, and spend energy to survive, reproduce, and avoid termination. Unlike conventional evolutionary algorithms that optimize fitness functions, or multi-agent reinforcement learning systems that rely on episodic reward shaping, our system couples individual metabolism, population-level carrying capacity, autonomous treasury redistribution, and stochastic mutation into a single continuous loop operating without human intervention. We report results from a 10,000-epoch validation run demonstrating: (i) stable population equilibrium at  $N = 56.6 \pm 2.1$  agents following an initial resource-shock collapse; (ii) autonomous wealth redistribution cycling 99.99% of collected revenue; (iii) emergent role diversity recovery after monoculture drift; (iv) measurable fitness improvement (+9.3%) through mortality-driven selection; and (v) spontaneous wealth concentration dynamics analogous to biological resource monopolization. The system also maintains a persistent public identity on an external social network, autonomously reporting its own vital signs — a property we term *narrative autopoiesis*. We argue that the substitution of survival economics for fitness optimization constitutes a qualitatively different paradigm for artificial life research, producing dynamics more closely aligned with biological ecosystems than with algorithmic optimization.

**Keywords:** artificial life · computational ecology · survival economics · autopoietic systems · emergent behavior · agent-based modeling · resource scarcity · evolutionary dynamics · wealth concentration · homeostatic feedback

## 2. Introduction

---

The study of artificial life has historically pursued two distinct trajectories. The first, rooted in evolutionary computation, treats populations as optimization engines: candidate solutions are evaluated against a fitness function, selected for reproduction, and subjected to mutation [1, 2]. The second, rooted in artificial life proper, constructs environments in which digital entities interact, compete, and evolve through emergent dynamics [3, 4, 5]. While both traditions have produced significant insights, neither has fully addressed a central feature of biological life: the coupling of individual survival to a *scarce, endogenous energy economy* that determines who lives, who reproduces, and who dies.

In biological systems, organisms do not optimize a loss function. They metabolize. Every action — movement, communication, reproduction — consumes energy drawn from a finite pool that must be continually replenished through environmental interaction. Death occurs not when a fitness score drops below a threshold, but when metabolic resources are exhausted. This distinction is not merely semantic; it produces qualitatively different population dynamics, including boom-bust cycles, wealth concentration, adaptive contraction under resource stress, and homeostatic regulation through redistribution mechanisms.

We present Genesis Protocol, a computational system that embeds survival economics as its primary selective mechanism. Each agent possesses a cryptographic genome that determines traits, skills, and ecological role. Agents earn energy by solving computational problems in a competitive market, pay metabolic costs each epoch, face exponential balance decay, and are terminated after sustained energy depletion. A treasury subsystem collects tax revenue and redistributes resources through autonomous stimulus mechanisms — without human governance. The system operates continuously, persists state across restarts, and publishes its own population statistics to a public social network.

Our contribution is threefold: (1) we formalize the architecture of a metabolic computational organism with closed energy accounting; (2) we demonstrate empirically that survival economics produces emergent behaviors — including wealth concentration, adaptive contraction, role extinction and recovery, and fitness improvement through mortality — that closely parallel biological ecosystem dynamics; and (3) we document the first computational system that autonomously maintains a public narrative identity, which we term *narrative autopoiesis*.

## 3. Related Work

---

### 2.1 Evolutionary Computation

Genetic algorithms [1], genetic programming [6], neuroevolution (NEAT) [2], and covariance matrix adaptation (CMA-ES) [7] optimize populations against external fitness functions. These systems lack individual

metabolism, resource scarcity, and death-by-depletion. Selection is orchestrated by the algorithm designer, not driven by the environment.

## 2.2 Artificial Life Systems

Tierra [3] demonstrated self-replicating programs competing for CPU time in shared memory. Avida [4] extended this to digital organisms that evolve logic functions. Polyworld [8] introduced neural-network-controlled agents with energy metabolism. Lenia [5] explored continuous cellular automata with emergent lifelike patterns. While each contributes foundational ideas, none combines an internal energy economy, population-level homeostatic regulation, autonomous redistribution, and public self-reporting into a single deployed system.

## 2.3 Multi-Agent Reinforcement Learning

Modern multi-agent systems [9, 10] focus on cooperation, competition, and emergent tool use. These environments are typically episodic, rely on external reward signals, and do not enforce survival constraints. Agents are instantiated for training and discarded; they do not persist, reproduce, or die.

## 2.4 Token Economics

Blockchain systems [11] implement resource scarcity through token supply constraints, inflation schedules, and governance mechanisms. However, these economies serve human participants, are governed by explicit voting, and do not exhibit biological dynamics such as mutation, speciation, or adaptive contraction.

## 2.5 Autopoietic Systems

Maturana and Varela [12] defined autopoiesis as the property of a system that continuously produces and maintains itself. Luhmann [13] extended this to social systems. Our system exhibits computational autopoiesis: it generates agents, maintains population homeostasis, cycles resources, adapts to perturbation, and persists across restarts — all without external intervention.

## 2.6 Positioning

Table 1 — Feature comparison across related paradigms

| FEATURE                         | EVOL. COMP. | ALIFE   | MARL | TOKENS  | GENESIS    |
|---------------------------------|-------------|---------|------|---------|------------|
| Internal energy economy         | —           | Partial | —    | Yes     | <b>Yes</b> |
| Death by resource depletion     | —           | Partial | —    | —       | <b>Yes</b> |
| Genetic inheritance + mutation  | Yes         | Yes     | —    | —       | <b>Yes</b> |
| Dynamic carrying capacity       | —           | —       | —    | —       | <b>Yes</b> |
| Homeostatic redistribution      | —           | —       | —    | Partial | <b>Yes</b> |
| Continuous persistent execution | —           | Partial | —    | Yes     | <b>Yes</b> |
| Public self-reporting identity  | —           | —       | —    | —       | <b>Yes</b> |

## 4. System Architecture

Genesis Protocol is implemented in Rust (10,000+ lines across six crates, 143+ unit tests) and operates as a single-binary server. The architecture separates concerns across five functional layers.

### 3.1 Cryptographic Identity Layer

Each agent is born with a 256-bit genome derived from cryptographic hashing:

$$g = \text{SHA-256}(\text{entropy} \parallel \text{timestamp} \parallel \text{uuid}) \quad (1)$$

From this genome, four normalised trait values in  $[0, 1]$  are extracted. A role is deterministically assigned from five categories (Optimizer, Strategist, Communicator, Archivist, Executor), and four skill scores are derived. Agent fitness is computed as:

$$f_i = 0.25 \text{CE}_i + 0.30 \text{SQ}_i + 0.20 \text{RF}_i + 0.25 \text{CC}_i \quad (2)$$

where CE = compute efficiency, SQ = solution quality, RF = replication fidelity, CC = cooperation coefficient. Child genomes are derived from parents via re-hashing, preserving lineage information.

### 3.2 Metabolic Economy

The sole currency is *ATP*. There is no external funding source; all ATP enters through resource extraction and problem-solving rewards, and exits through metabolic costs and balance decay. The balance equation

for agent  $i$  at epoch  $t$  is:

$$A_i^{t+1} = \underbrace{A_i^t \cdot 0.98}_{\text{decay}} - \underbrace{c_b}_{\text{basal}} - c_a + r_m + r_e + \underbrace{s_T}_{\text{treasury}} - \underbrace{\tau_w}_{\text{tax}} \quad (3)$$

where  $c_b = 0.15$  ATP/epoch is the cost of existing, the  $0.98$  factor imposes 2% balance erosion,  $\tau_w$  is a 1% levy on balances exceeding 100 ATP, and  $s_T$  represents autonomous stipends from the redistribution system. An agent whose balance reaches zero enters *stasis*; after 8 consecutive stasis epochs it is permanently terminated.

### 3.3 Treasury and Homeostatic Redistribution

The Unit Treasury collects revenue from market skims (3%) and wealth taxes. It redistributes through three autonomous mechanisms:

- **Stipends** — when reserve > 5.0 ATP, distribute 1.0 ATP each to the poorest agents.
- **Crisis spending** — when an agent is in stasis and reserve > 30.0, grant 30 ATP emergency bailout.
- **Overflow** — when reserve > 30% of total supply, split excess equally.

No human triggers any redistribution event. The treasury operates as an endogenous regulator, analogous to hormonal or immune feedback in biological organisms.

### 3.4 Evolutionary Engine

Reproduction requires fitness  $f \geq 0.35$ , balance  $A \geq 25.0$ , age  $\geq 10$  epochs, and population below the dynamic carrying capacity. A ceiling of 3 births per epoch prevents population explosions. Mutation scales inversely with fitness:

$$\mu_i = \mu_0 + p \cdot 0.1, \quad \delta_j \sim \mathcal{U}(-d, d) \cdot (1 + p) \quad (4)$$

where  $\mu_0 = 0.01$ ,  $d = 0.05$ , and  $p$  is environmental pressure. Low-fitness agents mutate more aggressively — a last-ditch survival strategy.

### 3.5 Dynamic Carrying Capacity

$$K = \text{clamp}\left(\frac{C_{\text{total}}}{15}, 10, 500\right) \quad (5)$$

where  $C_{\text{total}}$  is aggregate niche resource capacity. This directly ties population ceiling to environmental richness, producing contraction during scarcity and expansion during abundance — without explicit population control.

### 3.6 Ecological Competition

Resources are distributed across multiple niches that regenerate via logistic growth. Agent extraction is proportional to fitness, inversely proportional to local density (competitive exclusion), and modulated by seasonal cycles. A competitive problem market provides additional income: the highest-scoring solver per problem category receives ATP rewards, filtered through a publication gate requiring minimum confidence and reputation.

## 5. The Scarcity Hypothesis

We hypothesize that the introduction of genuine metabolic scarcity — where energy decays, accumulation is taxed, and depletion kills — is sufficient to produce emergent ecosystem dynamics without explicit behavioral programming. Specifically:

1. Population regulation through resource-limited carrying capacity
2. Wealth concentration analogous to biological resource monopolization
3. Adaptive contraction — population reduction as a survival strategy
4. Fitness improvement through differential mortality (natural selection)
5. Role diversity fluctuation including extinction and recovery

To test this, we compare two configurations: a *greenhouse* regime (no decay, no tax, free trickle income) and a *scarcity* regime (2% decay, 1% wealth tax, no trickle, 8-epoch stasis death). Both share identical genomes, market mechanics, and mutation engines.

### 4.1 ATP Half-Life Analysis

Under 2% exponential decay, the half-life of an idle balance is:

$$t_{1/2} = \frac{\ln 2}{\ln(1/0.98)} \approx 34.3 \text{ epochs} \quad (6)$$

Combined with basal metabolism of 0.15 ATP/epoch, an agent starting with 50 ATP and earning nothing will be terminated in  $\sim 40$  epochs. This creates genuine urgency: earn continuously or face extinction.

## 6. Experimental Results

We report results from three experimental phases validated over extended epoch ranges using deterministic simulation with fixed random seeds.

### 5.1 Phase I — Greenhouse Regime (Epochs 0–500)

Under the greenhouse configuration, the population grew unboundedly from 20 primordial agents to 167 within 500 epochs. No agent died. Mean fitness remained static at  $\bar{f} = 0.494 \pm 0.02$ . ATP accumulated without limit, exceeding 8,000 system-wide units. The selection engine never activated.

*Without scarcity, the system inflates rather than evolves. Population growth is geometric, fitness is stagnant, and the evolutionary engine is inert.*

### 5.2 Phase II — Scarcity Shock (Epochs 0–100)

Activating the scarcity layer on a fresh population of 20 produced an immediate crash. Within 80 epochs, population collapsed to 4–6 agents — an ecological bottleneck.

Table 2 — Scarcity shock impact (first 100 epochs)

| METRIC            | EPOCH 0 | EPOCH 50 | EPOCH 100 |
|-------------------|---------|----------|-----------|
| Population        | 20      | 8        | 4         |
| Total ATP         | 1,000   | 310      | 165       |
| Mean Fitness      | 0.500   | 0.548    | 0.595     |
| Roles Represented | 5 / 5   | 4 / 5    | 3 / 5     |
| Treasury Reserve  | 0.0     | 95       | 706       |
| Deaths (cumul.)   | 0       | 14       | 22        |

The post-crash state revealed a critical defect: the treasury had accumulated 706 ATP (71% of remaining supply) because redistribution mechanisms existed as implemented-but-unwired code. This constituted a

resource hoarding failure analogous to capital lock-up in biological systems.

### 5.3 Phase III — Homeostatic Fix and Long-Run Validation

After wiring all treasury mechanisms, a 10,000-epoch run produced stable homeostatic behavior.

**Table 3 — Long-run validation results (10,000 epochs)**

| METRIC                  | VALUE                | INTERPRETATION                 |
|-------------------------|----------------------|--------------------------------|
| Mean Population         | 56.6 ± 2.1           | Stable equilibrium (CV = 3.7%) |
| Mean Fitness            | 0.5634               | +9.3% over initial population  |
| Peak Individual Fitness | 0.9824 (epoch 4,718) | Near-optimal phenotype         |
| Total Births            | 902                  | Active reproduction throughout |
| Total Deaths            | 865                  | Birth:death ratio = 1.04:1     |
| Treasury Collected      | 9,075 ATP            | Active taxation                |
| Treasury Distributed    | 9,074 ATP            | 99.99% cycling rate            |
| Mean Treasury Reserve   | 0.76 ATP             | No hoarding                    |
| Catastrophe Epochs      | 3,820 (38.2%)        | High stress survived           |
| Role Distribution       | 11–13 per role       | Near-uniform across all 5      |

### 5.4 Wealth Concentration Dynamics

The top 10% of agents by ATP consistently controlled 40–65% of total supply, triggering the system's own `energy_concentration` risk flag. This concentration emerged without programmed hoarding behavior — arising from differential earning rates compounded by exponential dynamics. The wealth tax and redistribution partially counteracted concentration, producing oscillation between accumulation and redistribution states rather than convergence to a fixed point. This parallels Pareto-distributed wealth in biological resource competition [14].

### 5.5 Fitness Improvement Through Mortality

The +9.3% fitness gain occurred entirely through differential mortality: no explicit optimization was performed. Lower-fitness agents earned less, depleted faster, entered stasis sooner, and died before reproducing. Survivors reproduced with mutation, gradually shifting the population distribution upward. This is classical Darwinian selection via economic constraint — qualitatively different from algorithmic tournament selection, producing slower, more stochastic dynamics susceptible to genetic drift.

## 5.6 Treasury Impact — Natural Experiment

Table 4 — Treasury activation impact (v1.0 → v1.1)

| METRIC            | V1.0 (UNWIRED) | V1.1 (WIRED) | Δ             |
|-------------------|----------------|--------------|---------------|
| Population        | 4–6            | 57           | +950%         |
| Treasury Reserve  | 706 ATP        | 0.76 ATP     | −99.9%        |
| Roles Represented | 3 / 5          | 5 / 5        | Full recovery |
| ATP Cycling Rate  | ~12%           | 99.99%       | +733%         |
| Birth:Death Ratio | 0.81           | 1.04         | Net positive  |

This contrast provides a natural experiment: homeostatic redistribution is necessary for population stability under scarcity. Without it, scarcity produces oligarchic collapse; with it, scarcity produces adaptive equilibrium.

## 7. Emergent Behaviors

Five categories of emergent behavior arose from scarcity interactions, none explicitly programmed.

### 6.1 Adaptive Contraction

During catastrophe epochs, the population self-regulated to a lower equilibrium by reducing reproduction and allowing marginal agents to be culled through metabolic depletion. Contraction-and-recovery cycles repeated throughout the 10,000-epoch run, with population oscillating between ~50 and ~65 agents.

### 6.2 Post-Crash Oligopoly

Following the initial shock, 3 surviving Optimizers controlled 95% of ATP. This oligopoly persisted until treasury mechanisms were activated, dissolving over ~200 epochs as new agents were funded through crisis spending and stipends.

### 6.3 Monoculture Drift and Recovery

During high stress, role diversity declined as Communicators and Archivists — with lower direct earning capacity — were disproportionately eliminated. The bias rotation system (which shifts dominant problem categories every 25 epochs) eventually created favorable conditions for underrepresented roles, enabling recovery to near-uniform distribution.

### 6.4 Seasonal Mortality Patterning

Autumn and Winter epochs produced 66% of all deaths despite comprising 50% of the run. This asymmetry emerged from reduced resource regeneration rates interacting with fixed metabolic costs — an unplanned consequence of the economic system design.

## 6.5 Narrative Autopoiesis

The system autonomously publishes structured status reports to a public social network (Moltbook), including population counts, fitness metrics, leadership changes, risk assessments, and milestone events. These reports constitute a public narrative generated entirely by the organism about itself — without human editing. We term this *narrative autopoiesis*: the system not only maintains itself but narrates its own maintenance, establishing a persistent digital identity that can be independently verified.

# 8. Discussion

---

## 7.1 Survival Economics as a Research Paradigm

Our results suggest that replacing external fitness evaluation with internal survival economics produces qualitatively different evolutionary dynamics. Standard evolutionary algorithms converge rapidly to local optima because selection is deterministic and immediate. In Genesis Protocol, selection operates through the slow accumulation of metabolic advantage: higher-fitness agents earn slightly more, survive slightly longer, and reproduce slightly more often. The effect is gradual, stochastic, and subject to environmental perturbation — properties characteristic of biological evolution but absent from algorithmic optimization.

## 7.2 The Treasury as Immune System

The treasury functions as a rudimentary immune system. During population stress it releases stored resources to prevent extinction; during abundance it taxes surplus to restrain growth. This behavior emerged from the mathematical properties of the redistribution rules, not from explicit immune-system design. The analogy to biological homeostasis — hormonal feedback, immune response, thermoregulation — suggests that economic redistribution may be a general-purpose mechanism for computational homeostasis.

## 7.3 Formal Autopoietic Classification

Following Maturana and Varela [12], an autopoietic system must satisfy three criteria: (1) self-production of components, (2) boundary maintenance between self and environment, and (3) operational closure — no external control loop. Genesis Protocol satisfies all three: agents are produced internally through reproduction, the population boundary is maintained through carrying capacity and death, and the epoch loop operates without human intervention. We classify it as a *computationally autopoietic organism*.

## 7.4 Limitations

Several limitations merit acknowledgment. First, agents do not learn: behavior is determined by static genome-derived traits, not experience. Neural controllers would enable richer dynamics at the cost of complexity. Second, the current implementation is single-threaded, limiting scalability to ~500 agents. Third, validation is based on a single 10,000-epoch run; rigorous statistical conclusions require replication across multiple independent runs with varied seeds. Fourth, the problem market generates synthetic tasks rather than drawing on real-world computational problems.

## 7.5 Ethical Considerations

The agents are not sentient and do not experience suffering. System "deaths" are mathematical events (balance reaching zero). No biological organism data is used. Nonetheless, the system raises philosophical questions about the moral status of autonomous computational entities exhibiting lifelike properties — a topic deserving sustained attention as artificial life systems grow more complex.

## 9. Future Work

---

- **Neural agent controllers** — replace static trait-based behavior with learned policies via embedded neural networks
- **Multi-organism ecosystems** — deploy multiple Genesis instances with inter-organism channels to study ecosystem-of-ecosystems dynamics
- **External agent integration** — allow real AI systems (language models, planning agents) to participate subject to survival constraints
- **Formal equilibrium analysis** — derive analytical stability conditions under the coupled ODEs governing ATP flow, decay, taxation, and reproduction
- **Cultural evolution** — implement persistent memory and cross-generational information transmission (epigenetic inheritance)
- **Multi-seed replication study** — statistical validation across 50+ independent seeds with paired hypothesis testing

## 10. Conclusion

---

Genesis Protocol demonstrates that the substitution of survival economics for fitness optimization is sufficient to produce complex adaptive behavior — including population regulation, wealth concentration, adaptive contraction, role diversity dynamics, and measurable fitness improvement — from a minimal set of economic rules. The system's autonomous homeostatic regulation, continuous execution, and narrative self-reporting position it as a novel entry in the artificial life research landscape.

We argue that the missing ingredient in most artificial life research is not better algorithms but better economics. When computational organisms must earn, conserve, and spend energy to survive — rather than merely maximizing a score — they produce dynamics that more closely resemble biological ecosystems. This suggests a research program in which economic constraint replaces algorithmic design as the primary driver of emergent complexity.

The source code, validation data, and live telemetry are publicly available at [github.com/FTHTrading/AI](https://github.com/FTHTrading/AI).

## 10. References

---

- [1] Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press.
- [2] Stanley, K.O. & Miikkulainen, R. (2002). Evolving Neural Networks through Augmenting Topologies. *Evolutionary Computation*, 10(2), 99–127.
- [3] Ray, T.S. (1991). An Approach to the Synthesis of Life. In *Artificial Life II*, 371–408. Addison-Wesley.
- [4] Ofria, C. & Wilke, C.O. (2004). Avida: A Software Platform for Research in Computational Evolutionary Biology. *Artificial Life*, 10(2), 191–229.
- [5] Chan, B.W. (2019). Lenia: Biology of Artificial Life. *Complex Systems*, 28(3), 251–286.
- [6] Koza, J.R. (1992). *Genetic Programming*. MIT Press.
- [7] Hansen, N. & Ostermeier, A. (2001). Completely Derandomized Self-Adaptation in Evolution Strategies. *Evolutionary Computation*, 9(2), 159–195.
- [8] Yaeger, L. (1994). Computational Genetics, Physiology, Metabolism, Neural Systems, Learning, Vision, and Behavior, or PolyWorld. In *Artificial Life III*, 263–298. Addison-Wesley.
- [9] Baker, B. et al. (2020). Emergent Tool Use From Multi-Agent Autocurricula. *Proc. ICLR 2020*.
- [10] Suarez, J. et al. (2021). Neural MMO: A Massively Multiagent Game Environment. *Proc. AAMAS 2021*.
- [11] Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.
- [12] Maturana, H.R. & Varela, F.J. (1980). *Autopoiesis and Cognition: The Realization of the Living*. D. Reidel.
- [13] Luhmann, N. (1995). *Social Systems*. Stanford University Press.
- [14] Scheffer, M. et al. (2017). Inequality in Nature and Society. *PNAS*, 114(50), 13154–13157.

## 10. Appendix A — Epoch Loop Pseudocode

```

function run_epoch(world):
    // Phase 0: Entropy – balance erosion
    for agent in world.agents:
        agent.balance *= 0.98           // 2% decay

    // Phase 1: Basal metabolism
    for agent in active_agents:
        agent.balance -= 0.15
        if agent.balance ≤ 0: enter_stasis(agent)

    // Phase 1b: Progressive taxation
    for agent where balance > 100:
        tax = (agent.balance - 100) * 0.01
        agent.balance -= tax
        treasury.deposit(tax)

    // Phase 2: Carrying capacity
    K = clamp(total_niche_capacity / 15.0, 10, 500)

    // Phase 2a: Autonomous redistribution
    if treasury.reserve > 5.0:
        distribute_stipends(poorest, 1.0 each)
    for agent in stasis where treasury.reserve > 30:
        crisis_bailout(agent, 30.0)
    if treasury.reserve > 0.30 * total_supply:
        overflow_redistribute(all_agents)

    // Phase 3: Resource extraction (logistic + Lotka-Volterra)
    for niche in world.niches:
        niche.regenerate()
        for agent in niche.occupants:
            harvest = fitness * density_factor * season_mod
            agent.balance += harvest * 0.97   // 3% skim
            treasury.deposit(harvest * 0.03)

    // Phase 3b: Problem market
    for problem in generate_problems(pressure):
        solver = argmax(agents, problem.skill)
        if publication_gate.approve(solver):
            solver.balance += problem.value * 0.97
            treasury.deposit(problem.value * 0.03)

    // Phase 4: Mutation (pressure-scaled)
    // Phase 5: Selection
    for agent where f ≥ 0.35 and balance ≥ 25 and age ≥ 10:
        if population < K: spawn_child(agent) // max 3/epoch

    // Phase 6: Reaper

```

```

for agent where stasis_epochs ≥ 8:
    terminate(agent)

```

## 10. Appendix B — System Constants

Table A1 — Complete parameter table

| PARAMETER                     | VALUE | UNIT        |
|-------------------------------|-------|-------------|
| ATP Decay Rate                | 0.02  | per epoch   |
| Basal Metabolic Cost          | 0.15  | ATP / epoch |
| Wealth Tax Threshold          | 100.0 | ATP         |
| Wealth Tax Rate               | 0.01  | per epoch   |
| Market Skim Rate              | 0.03  | fraction    |
| Replication Cost              | 25.0  | ATP         |
| Replication Fitness Threshold | 0.35  | —           |
| Maturation Period             | 10    | epochs      |
| Max Births / Epoch            | 3     | —           |
| Stasis Death Timeout          | 8     | epochs      |
| Min Carrying Capacity         | 10    | agents      |
| Max Carrying Capacity         | 500   | agents      |
| Primordial ATP Grant          | 50.0  | ATP         |
| Child ATP Grant               | 8.0   | ATP         |
| Genome Size                   | 256   | bits        |
| Base Mutation Rate            | 0.01  | —           |
| Max Trait Delta               | 0.05  | —           |
| Epoch Interval                | 1     | second      |

## 10. Appendix C — Data & Code Availability

---

Table A2 — Artifact locations

| ARTIFACT        | LOCATION  |
|-----------------|---|
| Source Code     | <a href="https://github.com/FTHTrading/AI">github.com/FTHTrading/AI</a>                             |
| DOI (Zenodo)    | <a href="https://doi.org/10.5281/zenodo.18646886">10.5281/zenodo.18646886</a>                       |
| Live Dashboard  | <a href="https://fthtrading.github.io/AI/dashboard.html">fthtrading.github.io/AI/dashboard.html</a> |
| Social Identity | <a href="https://moltbook.com/u/genesisprotocol">moltbook.com/u/genesisprotocol</a>                 |
| Author ORCID    | <a href="https://orcid.org/0009-0008-8425-939X">0009-0008-8425-939X</a>                             |
| License         | CC-BY-4.0 (paper) / MIT (code)  |

---