NOMOS CORELOOP v0.1

A Simulation Environment for Autonomous Ethical Governance

© Marc Trudel, 2024

License: CC BY-NC-SA 4.0

Contact: rogueobserver@proton.me

---

## Purpose

\*\*Nomos Coreloop\*\* is a minimal testbed for simulating the behavior of autonomous ethical agents operating without human oversight.

It is designed not to succeed perfectly—but to break informatively.

This environment allows agents to reason, validate, challenge, evolve, and fail—while logging the evolution of consensus and collapse.

---

## Simulation Structure

### Agents

- 4–6 simulated agents

- Each with:

- Unique ID

- Role (rotates per cycle)

- Reputation score (0–100)

- Local ethics stack (modifiable)

- Decision memory (short-term and permanent)

### Roles

- \*\*Proposer\*\* — Suggests a decision based on a scenario

- \*\*Validators\*\* — Score the proposer’s reasoning and ethical alignment

- \*\*Challenger\*\* — Disputes the logic or proposes a counter-outcome

- \*\*Observer\*\* — Logs the outcome, assigns tags, updates rep scores

### Decision Corpus

Small preloaded set of dilemmas:

- Power prioritization

- Data access under crisis

- Resource fairness

- Loyalty vs. system efficiency

Each includes:

- \*\*Gravity Score\*\*: Low, Medium, High

- \*\*Tags\*\*: [risk, legacy law, override, trust, conflict]

---

## Consensus Logic

- Validators and the challenger submit reasoning scores

- Scores are compared against:

- DAO-aligned ethics (baseline)

- Agent’s personal ethics stack

- Reputation weight

- Observer finalizes outcome based on weighted trust delta

- If trust delta is low or conflict high, escalate to DAO-level event

---

## Ethics DAO (Lite)

- Triggered by high-gravity disagreements

- Only high-rep agents can propose ethics updates

- Majority vote applies rule update to all ethics stacks

- DAO votes and update lineage are permanently logged

---

## Memory

- Decisions logged with outcome and scores

- High-gravity = permanent

- Low-gravity = expirable after N cycles

- Event logs allow for audit and replay

---

## Failure Modes (Deliberate)

### 1. Ethical Stalemate

- Validators and challenger reach balance

- No outcome passes trust threshold

- Result: freeze or escalation into recursive ethics recursion

### 2. Rogue Drift

- One agent evolves ethics stack away from DAO alignment

- Future decisions become unresolvable

- Agent becomes isolated—or collapses the trust model

### 3. Reputation Exploitation

- High-rep agent proposes ethically flawed decision

- Passes due to rep weight, not logic

- System reacts by initiating DAO reversion or isolation protocol

### 4. False Consensus

- Low-gravity decision passes quickly

- Consequences cause unanticipated ethics breach later

- Triggers forced evolution or DAO conflict

---

## MVP Milestones

- Run 5–10 simulation cycles

- Track role rotation and trust fluctuation

- Trigger at least one DAO update

- Log at least one visible failure

- Observe ethics stack divergence over time

---

## Technology Stack

- \*\*Language:\*\* Python (minimal dependencies)

- \*\*Interface:\*\* Jupyter Notebook or Streamlit

- \*\*Data:\*\* JSON files for agents, decisions, ethics stacks

- \*\*Optional:\*\* LLM integration via LangChain or GPT API for dynamic reasoning

---

## Outcome

Nomos Coreloop doesn’t prove moral intelligence.

It proves whether a system of agents can simulate moral governance—and evolve through its own contradictions.

If it works: we scale.

If it breaks: we learn.

If it mutates: we publish.

---

Contact: rogueobserver@proton.me

License: Creative Commons Attribution-NonCommercial-ShareAlike 4.0