MTC TECHNICAL REPORT:

Stochastic and Empirical Evaluation of the CNVS Framework

1. Introduction to MTC Testing

This report documents the Architectural and Mathematical Evaluation Suite of Closed Native Verification Systems (CNVS).
The Monte Carlo tests (MTC) presented are not intended to replace the formal proof described in the Technical Document, nor do they claim to constitute a definitive formal proof of the Theory, but rather to project and measure its behavior on a large scale. The testing framework is structured to progressively move from pure statistical abstraction (assuming the axioms) to executable implementation in a software environment.

2. Purpose of Work

The objective of this testing campaign is twofold:
1. To evaluate, in the CNVS model, the ability to overcome the dependence on static compromise thresholds typical of traditional consensus protocols, without assuming direct equivalence between the respective threat models, which occupy separate domains.
2. To numerically verify the operational consistency of the projections derived from Theorem 4, showing that, under explicit assumptions and for the same C_{int} isolation, the reconstruction probability remains strongly suppressed even for large fractions of colluding nodes.

3. Materials and Methods

- **Execution environment:** Python 3.10+.
- **Stochastic Engines:** numpy.random library for probability sampling and generating uniform, one-to-one distributions. Most Monte Carlo projection tests use 100,000 iterations; Test 11 uses 20,000 iterations per parameter point due to its executable structural pipeline.
- **View:** matplotlib library for rendering decay logs.
- **Application Cryptography (for empirical testing):** SHA-256 hashing and native cryptographic pseudorandom modules, finite-field modular arithmetic (PRIME = 1,000,003) for implementing the nonlinear polynomial engine.

4. Types of Tests

The analysis is divided into four incremental categories:

1. **Statistical Projection Tests (4 Tests):** Monte Carlo engines that simulate network behavior assuming the formal validity of the CNVS equations. They measure the impact of the variables q, m, and h_{min}.
2. **Interactive Statistical Projection Tests (3 Tests):** Dynamic HTML/JS-based dashboards for real-time visualization of risk surfaces.
3. **Minimum Value Measurement Test mmin (1 Test):** Optimization algorithm for calculating the minimum critical threshold of fragments needed to ensure system safety below a specific threshold τ.
4. **Empirical Demonstrative Tests (3 Tests):**
    - A structural pedagogical model demonstrating the separation between authentication and truth (V_L vs V_G).
    - A full structural-semantic execution model instantiating hidden invariant binding, topological refresh, and finite-field geometric invariants (C_{int}) under mass peripheral compromise.
    - An executable fragmentation-sensitivity model evaluating how increasing critical fragmentation cardinality m affects unauthorized V_G acceptance under severe peripheral compromise.

5. Test Analysis 1

Test Name: Test 1 - Statistical Projection of Systemic Reconstruction under Dependent Collusion (Min-Entropy Variation).

Test Purpose:

The test evaluates the systemic failure rate (when the adversary manages to circumvent the Global Veto) as a function of the percentage of colluding nodes (q). It computes 100,000 stochastic iterations using the formula $\mathbb{P}(Rec^*) \leq p_{comp}(t)^{m(t)}$, varying only the strength of the Residual Min-Entropy (h_{min}), i.e., the cryptographic uncertainty the adversary faces when trying to guess fragments it doesn't physically possess.

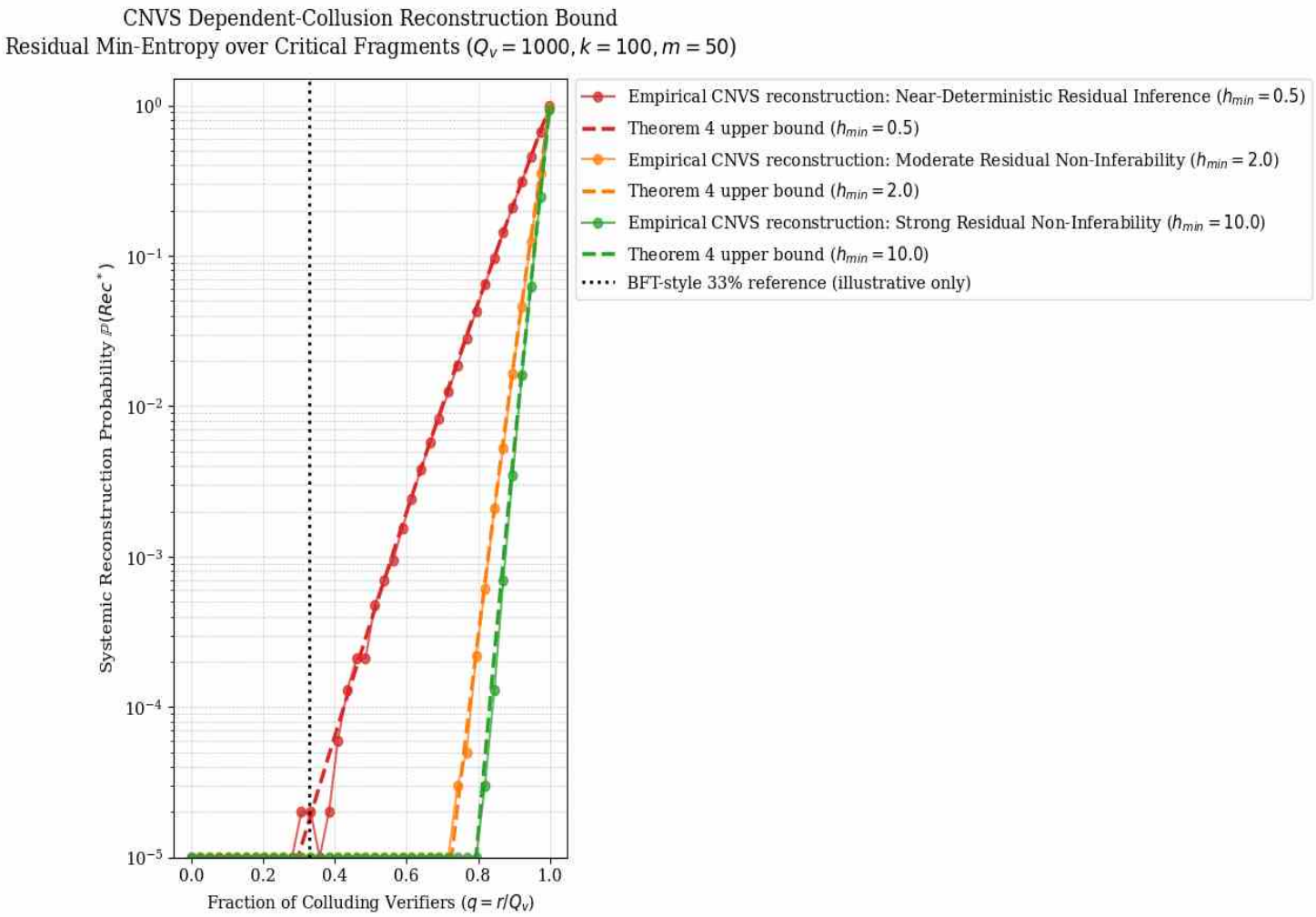
Formal Assumptions (Rules of Engagement):

This is a statistical projection test, not a physical empirical test.
Therefore, it operates under the following strict assumptions:

- **Assumption 1 (Axiomatic Validity):** Theorem 4 is assumed to be mathematically correct. The code does not compute the actual encryption of the data, but rather computes the probabilities derived from the formula.

- **Assumption 2 (Injective Assignment):** Each terminal fragment is assigned to only one node (1 fragment = 1 verifier).
- **Assumption 3 (Uniformity of the inference bound):** We assume that, for each missing fragment, the adversary has a maximum probability of success in guessing the data equal to $p_{inf} \leq 2^{-h\_min}$.
- **Assumption 4 (Global Veto Condition):** It is assumed that for the opponent to win, he or she must possess or successfully infer 100% of the m critical fragments.
  The lack of a single critical fragment triggers the Veto.

Output Graph:



Technical Commentary on the Data:

The graph plots the Reconstruction Probability $\mathbb{P}(Rec^*)$ on the logarithmic Y-axis against the fraction of colluding nodes q on the X-axis (from 0.0 to 1.0). The fixed network parameters are the verifier universe (Q_v = 1000), active shards (k = 100), and the critical shield (m = 50).

(Note: The "Empirical" in the image legend refers to the statistical output simulated by the Monte Carlo library versus pure theoretical calculation, not a real-world application run.)

1. **The Black Dashed Line (BFT Limit 33%):** This vertical line marks the critical threshold in classical BFT models. Above q = 0.33, many classical BFT models lose the standard Byzantine tolerance guarantees. In CNVS, at this threshold, the attacker's success probability for all three curves appears rigidly anchored to 10^{-5}.
2. **Red Curve (Low Min-Entropy, $h_{min} = 0.5$):** This curve represents the most challenging scenario for preserving the integrity of the CNVS. The fragments are highly predictable. The residual min-entropy margin is weak: $h_{min} = 0.5$ implies $p_{inf} \leq 2^{-0.5} \approx 0.707$, meaning that each missing critical fragment remains highly inferable under this pessimistic setting.
3. **Orange Curve (Moderate Entropy, $h_{min} = 2.0$):** With slightly higher uncertainty, the system exhibits remarkable resilience. The adversary's success probability remains extremely low until it physically corrupts a threshold τ close to 80% of the entire network.
4. **Green Curve (High Entropy, $h_{min} = 10.0$):** This is the expected behavior of the CNVS framework under optimal conditions. The curve demonstrates that, if epistemic isolation is intact, the adversary can corrupt thresholds of up to approximately 85% of the nodes without achieving a significant probability of forcing a global validation state within the system.
5. **Theorem-Simulation Convergence:** The point markers (the results computed from the 100,000 iterations of the simulator) overlap the dashed lines (the algebraic equation of Theorem 4).

Conclusions of Test 1:
The test shows that, if the axiomatic assumptions are met, in the CNVS model, the probability of falsification is decoupled from the percentage of compromised nodes alone. Fault tolerance is not governed by a fixed majority threshold alone, but a logarithmic curve that can be pushed asymptotically toward extremely high collusion threshold values by manipulating only the granularity (m) and data entropy (h_{min}) parameters.

6. Analysis of Test 2

Test Name: Test 2 - Statistical Projection (Stress Test) under Dynamic Erosion of the Residual Min-Entropy.

Test Purpose:

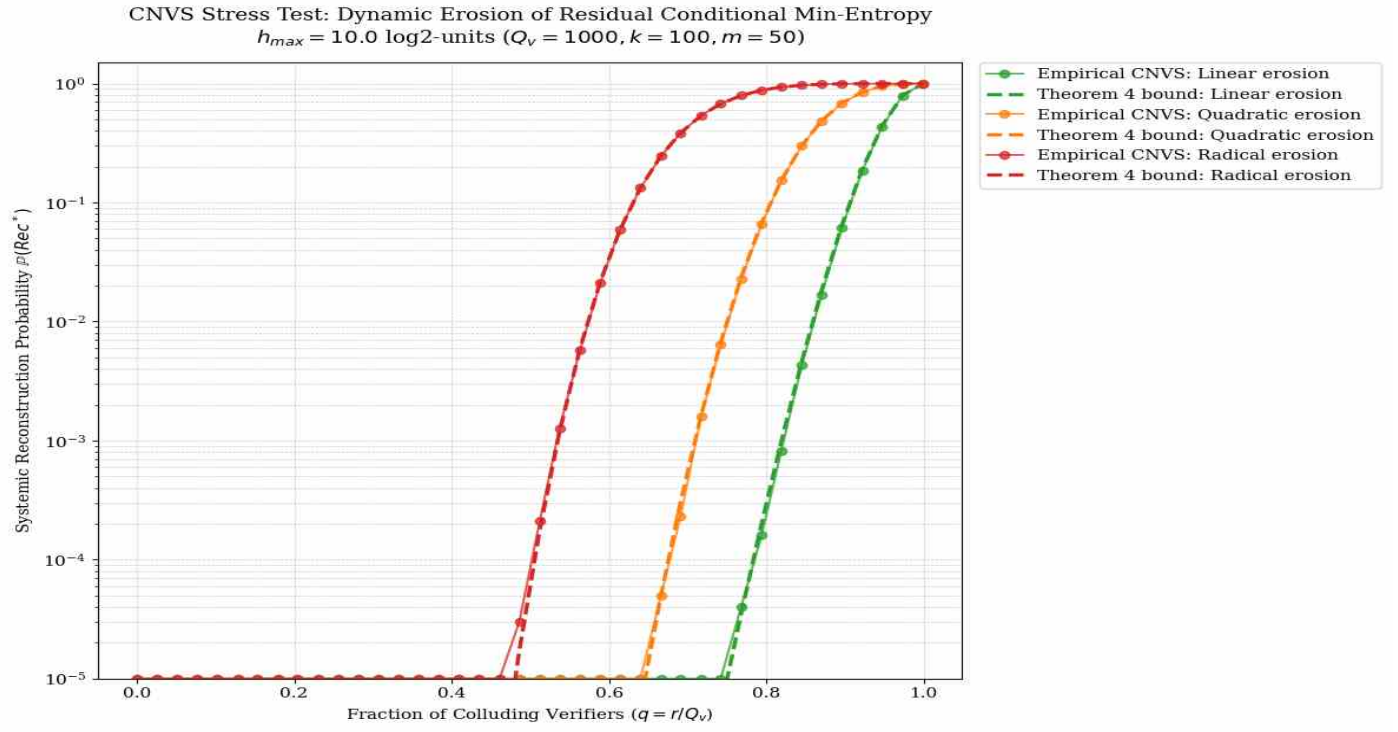
Test 2 serves as a formal stress test for the reconstruction layer of the CNVS framework. Unlike Test 1 (which evaluates the network under a constant entropy margin), this simulator models a dynamic decay of the epistemic advantage. The objective is to measure the resilience of the Global Veto when the proliferation of colluding nodes (q) causes a progressive and

accelerated loss of cryptographic entropy (h_{res}), simulating scenarios of massive metadata exfiltration.

Formal Assumptions (Rules of Engagement):
As with Test 1, the analysis is a statistical projection that assumes the mathematical validity of Theorem 4 ($\mathbb{P}(Rec^*) \leq p_{comp}(t)^{m(t)}$). The specific rules for this stress test are:

- **Assumption 1 (Entropy Baseline):** The system starts from an optimal and constant residual min-entropy margin equal to $h_{max} = 10.0$.
- **Assumption 2 (Heuristic Erosion Profiles):** Entropy attrition caused by the expansion of the opposing coalition is non-linear in nature, so it is intentionally forced through three mathematical stress profiles:
    - **Linear:** Entropy degrades directly proportional to the uncompromised network, $h_{max} \cdot (1 - q)$.
    - **Quadratic:** Degradation accelerates, simulating an aggravated metadata corruption, $h_{max} \cdot (1 - q)^2$.
    - **Radical:** A scenario in which degradation is maximum, topological isolation is minimum, and in which entropy deteriorates to the fourth power, $h_{max} \cdot (1 - q)^4$.
- **Assumption 3 (Dynamic Inference):** The upper bound of the adversarial inference probability pinf is not static, but is iteratively recalculated as a function of the eroded entropy $h_{res}(q)$.

Output Graph:



Technical Commentary on the Data:

The logarithmic scale plot plots the Systemic Reconstruction Probability versus the fraction of colluding nodes q, applying the three dynamic erosion profiles to the standard network parameters (Q_v = 1000, k = 100, m = 50).
(Methodological note: The term "Empirical CNVS" in the plot legend refers to the output of the Monte Carlo stochastic sampling, generated independently of the algebraic calculation of Theorem 4).

1. **Stochastic-Theoretical Overlap:** In all three erosion profiles, the point markers closely overlap the dashed curves within Monte Carlo resolution. This confirms numerical consistency between the Monte Carlo projection and the analytical bound under the selected nonlinear erosion profiles.
2. **Green Curve (Linear Erosion):** Under proportional metadata degradation, the CNVS architecture maintains significant resilience, with a probability of unauthorized reconstruction close to statistical zero (10^{-5}) up to a physical corruption of 75% of the nodes (q = 0.75). The system only begins to fail in the last quarter of the spectrum.
3. **Orange Curve (Quadratic Erosion):** By applying an accelerated entropy decay, the relational structure of the 50 critical fragments (m = 50) neutralizes the attack while keeping the risk of system breach still close to statistical zero up to a threshold close to 65% (q = 0.65).
4. **Red Curve (Radical Erosion):** In the worst-case scenario, where metadata collapses to the fourth power, significantly reducing the defender's epistemic advantage, under certain conditions, the framework still manages to demonstrate remarkable resilience. The curve rises from the bottom of the scale at around 45-50% corruption. This means that, even under conditions of significant cryptographic degradation, CNVS requires the attacker to physically control half the network before reaching a probability observable at the resolution of the Monte Carlo sampling used.

Test 2 Conclusions:
The dynamic erosion stress test shows that the CNVS architecture exhibits structural resilience under the selected assumptions. Fault tolerance depends not solely on data opacity (h_{min}), but on the geometric fragmentation itself (m). Even assuming massive metadata exfiltration (Radical profile), the attacker is forced to significantly exceed the failure limits of standard BFT systems (33%) before achieving a measurable success rate. Under the less aggressive erosion projections (Linear or Quadratic), the system develops significant resistance beyond the majority of nodes.

7. Analysis of Test 3

Test Name: Test 3 - Statistical Projection of Reconstruction Decay and Economic Penalty (Slashing) Model under Extreme Dependent Collusion.

Test Purpose:

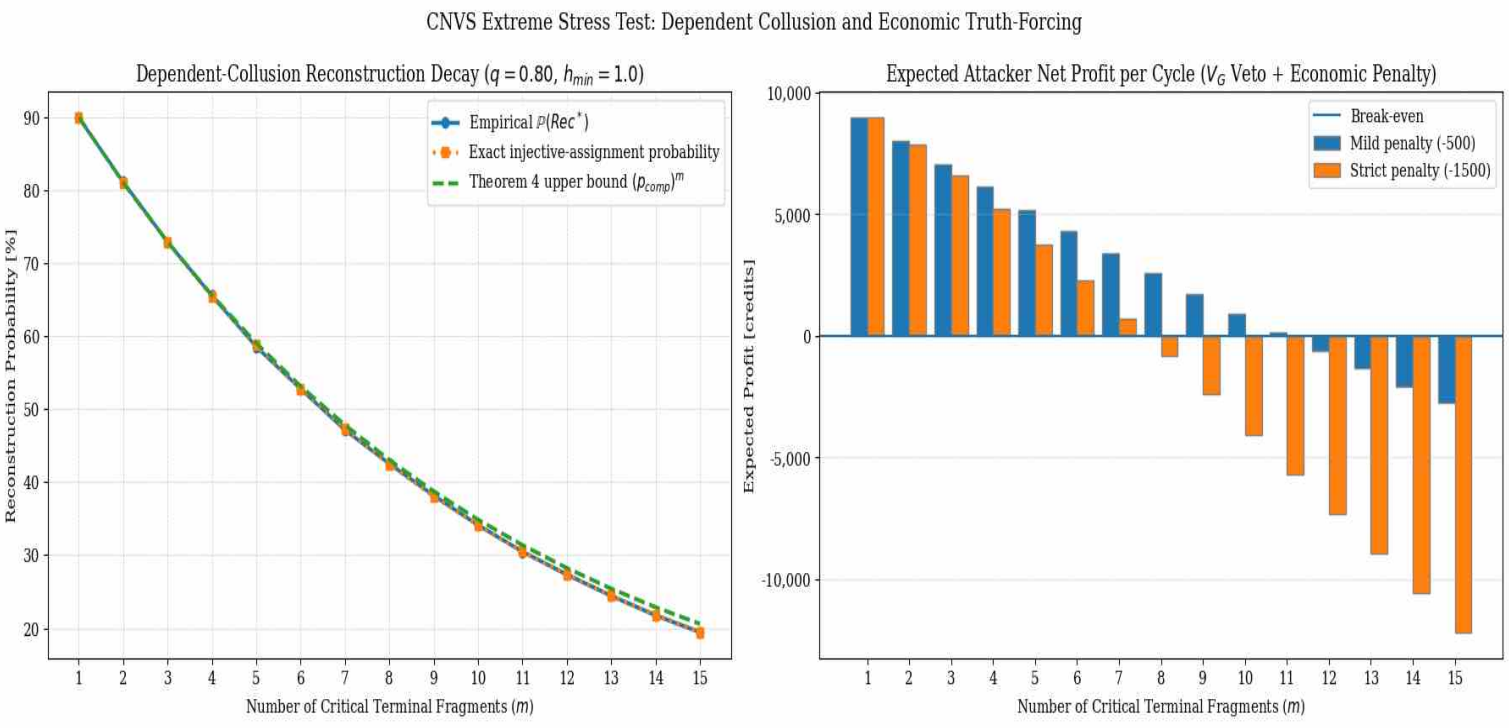
Test 3 projects the behavior of the CNVS framework by introducing an application implementation variable: the economic fault-tolerance model (Economic Truth-Forcing). The goal is to measure the System Reconstruction Probability in a scenario of severe topological compromise (80% of the network) and, simultaneously, calculate the Expected Value (EV) of an attacker's profit, demonstrating how the Global Veto translates into a strong financial disincentive.

Formal Assumptions (Rules of Engagement):

In addition to the assumption of validity of Theorem 4 and the injective assignment, this test operates under specific stress parameters:

- **Assumption 1 (Adversarial Physical Domain):** Fraction of colluding nodes set to $q = 0.80$. The attacker controls 80% of the infrastructure, a threshold that is well beyond common reference thresholds used in several consensus-based fault-tolerance models, noting that CNVS and consensus protocols operate under different threat models.
- **Assumption 2 (Low Epistemic Uncertainty):** Weak residual entropy margin, $h_{min} = 1.0$. The adversary has a 50% chance ($p_{inf} \leq 0.5$) of correctly inferring every single missing critical fragment.
- **Assumption 3 (Shield Variable m):** The number of total terminal fragments k is 50. The analysis observes the risk decay by progressively increasing the critical subset m from 1 to 15.
- **Assumption 4 (Economic Model and Slashing):**
    - Veto Override Reward (VG): +10,000 credits.
    - Penalty Mode: "Critical Only" (only nodes directly involved in the failure of the C_{int} Invariant suffer a funding cut).
    - Minor Penalty: -500 credits per node.
    - Severe Penalty: -1500 credits per node.

Output Graph:



Technical Commentary on the Data:

The visual panel is divided into two related projections.
(Methodological note: The "Empirical" label in the first graph refers to the statistical output simulated by Monte Carlo convergence over 100,000 iterations, which is compared to the analytical bound.)

1. Left Graph (Decay of Reconstruction Probability):

- The graph shows the decay of the probability of unauthorized reconstruction as the parameter m increases (X-axis).
- We observe an almost superimposable alignment between stochastic sampling, exact probability by injective assignment, and the theoretical upper bound expressed by Theorem 4.
- **Critical fact:** Despite the attacker controlling 80% of the network (q = 0.80) and enjoying high inference power (p_{inf} = 0.5), the probability of success drops from 90% (with m = 1) to just under 20% (with m = 15). The graph shows how the expansion of the relational surface (m) partially compensates for physical compromise under the selected parameters.

2. Right Graph (Attacker's Expected Net Profit):

- This histogram translates the probability of success into economic Expected Value (EV). As long as the EV is positive (bars above the zero line), the attack is rationally advantageous.
- As m increases, the frequency of Global Vetoes increases. Since each veto involves identifying and slashing the funds of colluding nodes detected in the invariant, the attacker's net profit progressively erodes.
- **Break-even Point:** With a severe penalty (-1500), the expected profit crosses the zero line at m = 8. With a mild penalty (-500), the cross occurs at m = 12.
- Beyond these points, the bars become negative. The system generates an estimated net loss for the opponent.

Conclusions of Test 3:

Test 3 assumes CNVS as a self-sustaining and self-defending cryptographic economic ecosystem. The geometric invariant, in addition to rejecting altered states, serves as a linchpin for the application of asymmetric economic penalties. Specifically, the framework shows how controlling 80% of the network's nodes does not translate into a sustainable attack vector; on the contrary, it exposes the adversary to a progressively unfavorable economic risk under the selected penalty parameters, a condition that can be exploited in the implementation by adjusting the ratio between m and the penalty multiplier.

8. Analysis of Test 4

Test Name: Test 4 - Statistical Projection of Topological Refresh and Sybil Purge (Iterative Fault-Injection Model).

Test Purpose:

Test 4 evaluates the resilience of the CNVS architecture in the temporal dimension. Unlike previous tests, which focused on the probability of spatial reconstruction at a single point in time t, this model projects the evolution of a continuous iterative attack. The goal is to measure the accumulation of topological metadata (M_S) and the decrease in the adversary's offensive capability by comparing a static (traditional) network topology with the ephemeral CNVS topology, governed by limited validation windows and structural refreshes.

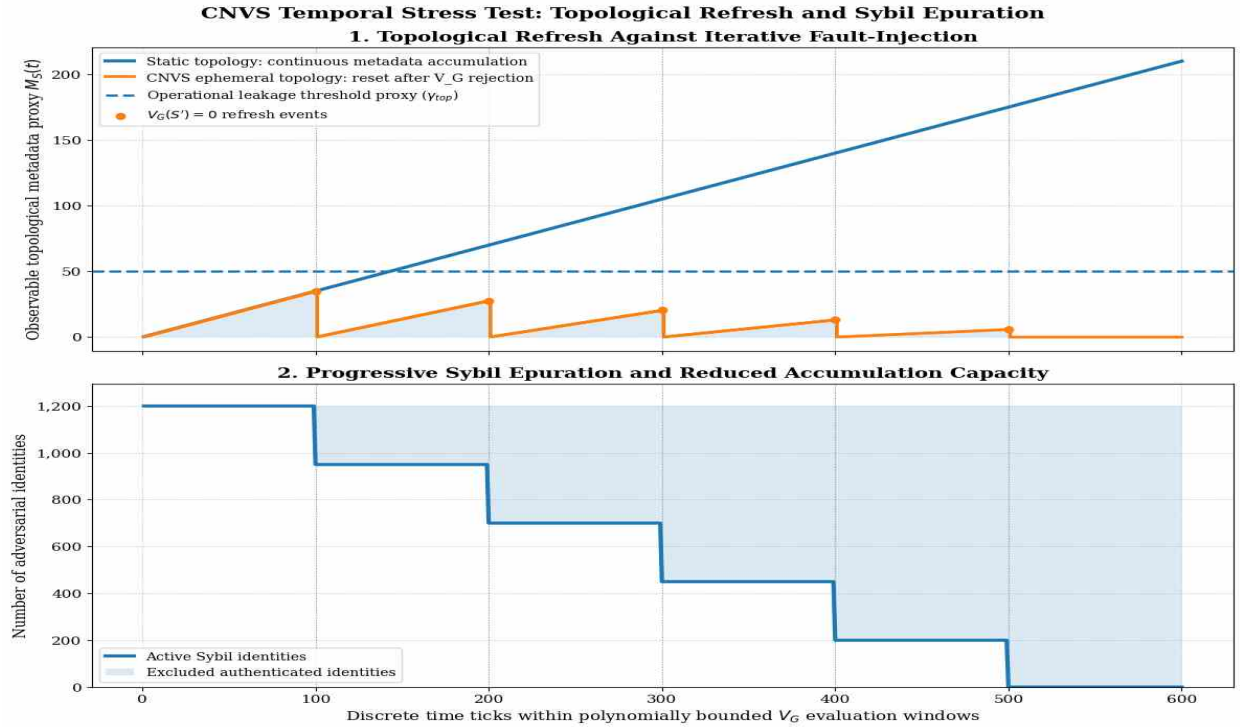
Formal Assumptions (Rules of Engagement):

The model operates according to the following formal architectural dynamics:

- **Assumption 1 (Static Baseline):** In a conventional distributed network (blue line), observable metadata accumulates linearly and continuously as long as the attack persists, inevitably exceeding the critical security threshold.

- **Assumption 2 (Bounded Evaluation Window - teval):** In the CNVS, the global validation function (V_G) does not wait forever. It resolves the transition within a predefined polynomial time window (set in the test at 100 discretionary ticks).
- **Assumption 3 (Topological Refresh):** A rejection by the Global Veto ($V_G = 0$) triggers an immediate reshuffling of the relational topology, reducing or invalidating the carry-over usefulness of metadata collected by the adversary during the current cycle.
- **Assumption 4 (Sybil Proxy Purge):** Authenticated corrupted identities, used by the attacker in the failed transition, are identified and expelled from the network (rate of 250 identities per cycle), reducing the adversary's accumulation power and leading to the progressive reduction of the adversary's ability to control the network.
- **Assumption 5 (Proxy Risk Threshold - γ_{top}):** Conventionally set at 50 units, it serves as a visual indicator of the limit beyond which the accumulation of metadata could compromise epistemic isolation in the adopted proxy model (Lemmas 9 and 10).

Output Graph:



Technical Commentary on the Data:

The visual panel is structured on two shared time axes (from 0 to 600 ticks).

1. Top Panel (Metadata Accumulation Dynamics):

- **The blue line (static topology)** illustrates the inherent vulnerability of a persistent system: the continuous accumulation of metadata crosses the critical threshold (γ_{top} = 50) around tick 140, leading to a violation of epistemic isolation.
- **The orange stepped line** (CNVS) shows a radically opposite behavior. Once the evaluation window limit is reached (t=100, 200, 300, etc.), the Global Veto first blocks the attack and then forces a topological and invariant refresh, bringing the adversarial knowledge back to the starting point (vertical line below the orange markers).
- **Critical data:** At no point in the simulation does the orange line approach the critical threshold. The maximum peak at tick 100 settles within a safe range, demonstrating the system's stability.

2. Bottom Panel (Enemy Population Decay):

- **The dark blue line** traces the volume of active Sybil identities, starting from an initial contingent of 1200 controlled nodes.
- At each *Refresh* event (coinciding with the vertical dashed lines), the system performs a "purge": The malicious population is progressively reduced.
- **Structural Correlation:** There is a direct mathematical connection between the bottom and top panels. As the number of active Sybils decreases (Panel 2), the adversary's ability to extract metadata slows. This is visible in Panel 1, where the slope of the orange segments (the accumulation rate) becomes less steep with each cycle, until it flattens completely at tick 500, when the adversary's population hits zero.

Conclusions of Test 4:

The test illustrates, under an idealized operational proxy model, how bounded evaluation windows, topological refresh, and authenticated-identity epuration can limit metadata accumulation during iterative fault-injection attempts. Time does not work in the attacker's favor. An attacker's iterative fault-injection strategy not only produces a "null result" (a simple failure), but also becomes counterproductive: each failed iteration burns adversary resources ("Purge"), reducing the carry-over usefulness of metadata acquired during the previous cycle. The CNVS architecture exhibits a behavior in which the network is not passively subjected to the attack, but progressively reduces the adversary's operational capacity in the simulated model.

9. Analysis of Test 5 (Category 3)

Test Name: Test 5 - Statistical Projection of Minimum Critical Fragmentation (m_{min}) under Asymmetric Topological Exposure.

Test Purpose:

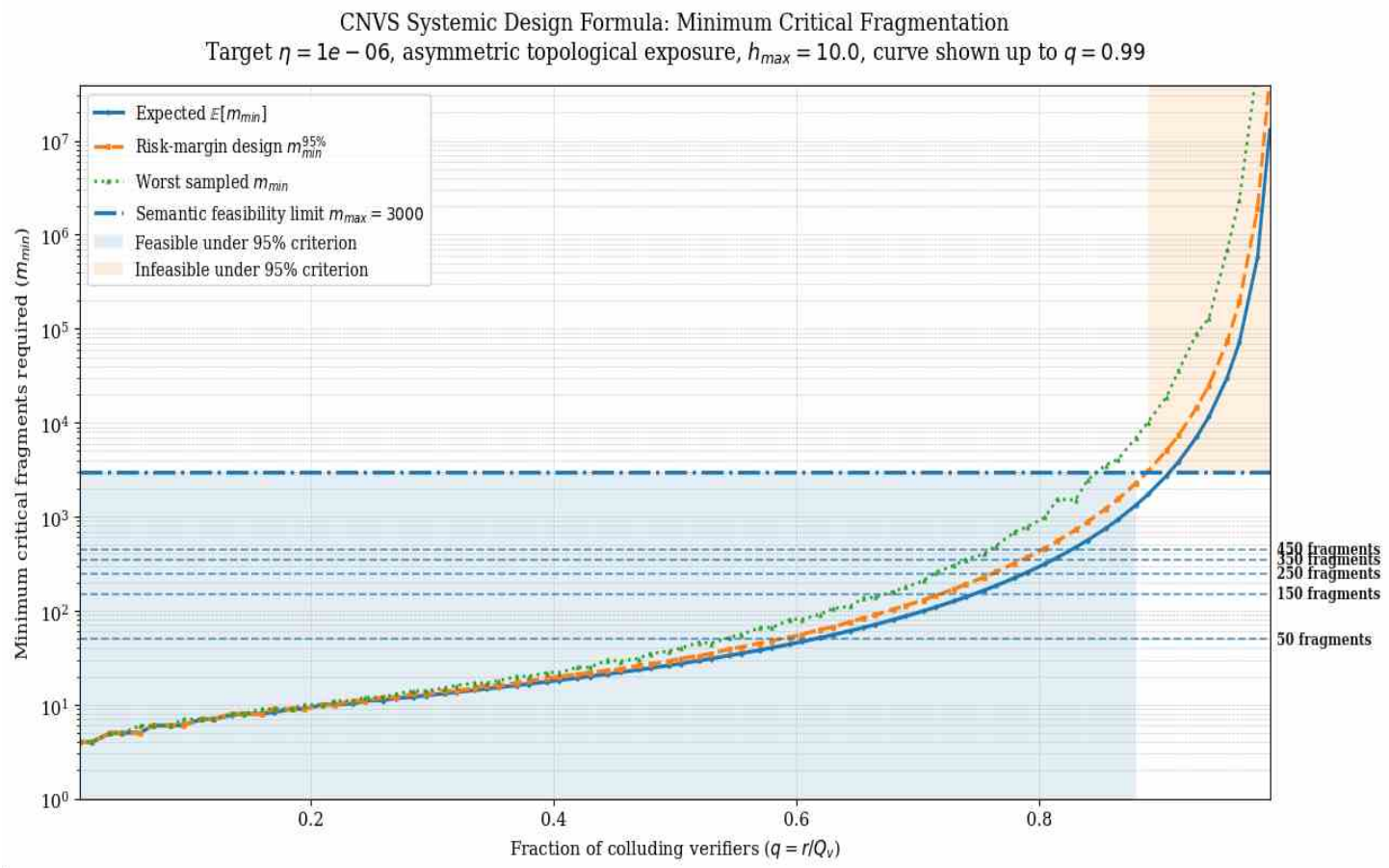
This test abandons the evaluation of the systemic failure rate (the subject of previous tests) to address an architectural engineering problem. The goal is to dynamically calculate the

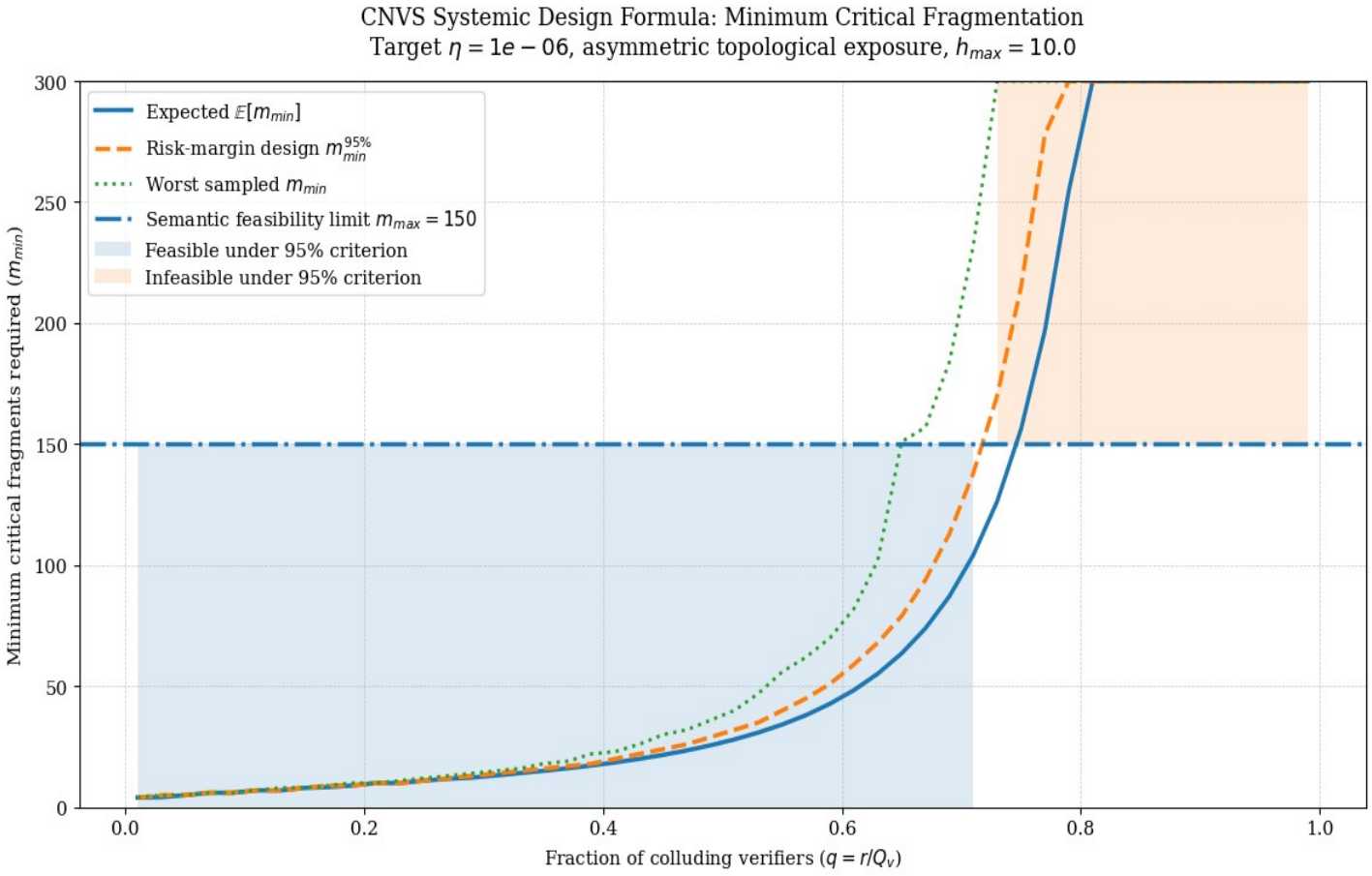
minimum threshold of critical fragments (m_{min}) that a CNVS network designer must instantiate to guarantee a predetermined safety target (Reconstruction Probability, here defined as $\eta \leq 10^{-6}$), as node corruption (q) increases.
The test employs the Systemic Design Formula (Eq. 42).

Formal Assumptions (Rules of Engagement):

- **Assumption 1 (Security Target η):** A strict risk tolerance is imposed: the probability that the adversary breaches the network must remain less than or equal to one in a million ($\eta \leq 10^{-6}$).
- **Assumption 2 (Asymmetric Topology):** Unlike a homogeneous linear compromise, the attacker compromises the infrastructure asymmetrically. This inhomogeneity is simulated using stochastic sampling weighted on a Dirichlet distribution ($\alpha = 1.0$).
- **Assumption 3 (Indirect Entropy Erosion):** The observable topological exposure (W_{top}) acquired by colluding nodes dynamically degrades the residual entropy (h_{res}) according to a quadratic profile. However, it does not directly alter the mathematical fraction of physically controlled nodes (q).
- **Assumption 4 (Semantic Feasibility Bound - m_{max}):** A practical feasibility bound ($m_{max} = 3000$) is imposed as a structural constraint on the recursive decomposition. As reported in the documentation, this bound constitutes the threshold value beyond which the decomposition causes the semantic meaning of the reference task to be lost, making it impossible for the verifier to execute the task.

Output Graph:





Technical Commentary on the Data:

The output is structured into two complementary views: the first analyzes the global asymptotic behavior on a logarithmic scale (up to m = 107), the second offers a linear cross-section of the application focused on high-density data ecosystems (limit m = 300). The plotted curves represent the Expected Value ($\mathbb{E}[m_{min}]$), the 95th percentile risk margin ($m_{min}^{95\%}$), and the sampled worst-case scenario.

1. Asymptotic Behavior Analysis (Graph 1 - Logarithmic Scale):

- The graph shows that the need for fragmentation grows in a controlled, sub-exponential fashion for much of the collusion spectrum, then experiences an asymptotic surge (vertical wall) as q approaches the critical threshold of 0.99. This is consistent with the mathematical bound that, if $p_{comp} \to 1$, $m_{min} \to \infty$.
- Given an extremely large semantic feasibility limit (m_{max} = 3000, blue mixed line), the light blue area shows that in practical implementation, it is possible to exploit the ^{violation} probability constrained by $\eta \leq 10^{-6}$ for node corruption rates close to 88-90% (evaluated using the 95th percentile safety criterion). Beyond this threshold, the system would require a fragmentation amount greater than 3000, entering the orange area of practical infeasibility (the 3000 threshold is arbitrary).

2. Practical Engineering Constraint Analysis (Graph 2 - Linear Scale):

- The second graph imposes a much more restrictive semantic limit for compact data transactions: m_{max} = 150 fragments.
- The divergence between the Expected Value (solid blue line) and the 95th percentile design curve (dashed orange line) is observed.
- Under this constraint, the system remains fairly secure ($\eta \leq 10^{-6}$) up to 72% network corruption. If collusion exceeds this threshold, the system would require more than 150 fragments set to the critical threshold m_{max} to maintain the premise, making the theory inapplicable in these extreme settings.

3. Impact of Asymmetric Exposure:

- The distance between the Expected Value curve and the "Worst Sampled" curve (dotted green line) quantifies the risk arising from topological asymmetry. Even in the case of an unfavorable concentration of metadata in the adversary's hands (Worst Sampled), the impact on the mmin parameter translates into a manageable deviation, without invalidating the model up to the identified tolerance thresholds.

Conclusions of Test 5:

The test graphically demonstrates the effect of Equation 42 as a formal architectural provisioning tool. The equation provides a computable provisioning parameter under explicit CNVS assumptions. Within the simulated provisioning model, by accepting a sufficient degree

of semantic granularity (increasing m_{min}), it is possible to manage the increase in physical corruption (q) while preserving network security.
The two graphs provide system architects with calibration curves: a payload composed of 150 covertly constrained critical fragments would be sufficient to survive a 72% compromise rate.

10. Analysis of Test 6 (Category 2: Interactive Models)

Test Name: Test 6 - Interactive Projection of Expected Value and Economic Tolerance under Global Veto.
Test Purpose:

Test 6 translates the static projection (Test 3) into an interactive application dashboard (HTML/JS). The goal is to provide the user with a real-time provisioning tool. By adjusting the sliders for adversary strength (q, h_{min}) and financial weights (Reward, Penalty), the user observes the shift of the break-even point on the histogram, thus demonstrating how the CNVS framework can be calibrated to verify its economic effects on the attacker (Truth-Forcing).

Formal Assumptions (Rules of Engagement):

- **Assumption 1 (Projection of Theorem 4):** The JavaScript algorithm natively executes the equation pcomp = q + (1 - q) 2^{-h_min} and raises the result by m to derive the probability of success ($\mathbb{P}(Rec^*)$).
- **Assumption 2 (Calculation of Expected Value):** The economic model is based on the Expected Value (EV) equation:

  $EV = [\mathbb{P}(Rec^*) \cdot Reward] + [(1 - \mathbb{P}(Rec^*)) \cdot (Penalty \cdot Exposed\ Nodes)]$.
- **Assumption 3 (Estimation of Exposed Nodes):** In case of rejection of the Invariant ($V_G = 0$), the number of colluding nodes identified and slashed is statistically estimated as m q, representing the share of critical fragments held by the opposing coalition.

Technical Commentary on Dynamics:

The interactive model shows the sensitivity of the framework to relational variables:

1. **Impact of the Adversarial Fraction (q):** By increasing the q slider to extreme values (e.g., 0.90), the histogram shows that corruption remains theoretically profitable for the lower values of m (green bars). However, simply scrolling the x-axis shows that introducing greater fragmentation (increasing m) drags the Expected Value below zero (red bars), reversing the trend.
2. **Financial Risk Calibration:** The dashboard shows the user that, under the specified assumptions, an "infinite penalty" is not necessary to discourage attack. By maintaining a very high reward and a relatively modest penalty per node, the reduction in probability generated by increasing m allows the weight of the penalties to outweigh the low probability of profit.

3. **The Knowledge Boundary (h_{min}):** Reducing entropy to zero ($h_{min} = 0$) transforms the system into a purely transparent domain where the adversary infers every missing fragment ($p_{inf} = 1$). In this extreme scenario where epistemic isolation is compromised, the Expected Value remains permanently green. The test's interactivity visually demonstrates the importance of the epistemic isolation condition.

Conclusions of Test 6:

The interactive application demonstrates that the CNVS security assumptions can be constrained by an adjustable economic equation. This allows the cybersecurity paradigm to be adjusted based on asymmetric crypto-economic security: the user can leverage the adversary's financial loss surface during the implementation phase. The dashboard shows how the model can transform a hypothetical attack, under selected parameters, into a counterproductive investment.

11. Analysis of Test 7 (Category 2)

Test Name: Test 7 - Interactive Projection of Probabilistic Decay and Epistemic Isolation.

Test Purpose:

Test 7 is an interactive exploration of the fundamental CNVS inequality, $\mathbb{P}(Rec^{*}) \leq [q + (1 - q)\, 2^{-h\_min}]^{m}$. It allows us to visually assess the isolated impact of two systemic defense variables: relational exposure (Fragmentation and Critical Fragments) and data opacity (Residual Entropy, hmin), by measuring the reaction of the risk curve with respect to the fraction of adversarial nodes (q) on the X-axis.

Formal Assumptions (Rules of Engagement):

- **Assumption 1 (Generating Equation):** The HTML/JS dashboard performs the asymptotic reconstruction calculation by plotting the full spectrum of q (from 0.00 to 0.99) for each micro-variation of the sliders.
- **Assumption 2 (BFT Reference):** The test adopts a vertical visual marker anchored at q = 0.33 as a reference. This represents the formal limit of traditional distributed systems based on majority consensus. The marker serves to scale the deviation of the geometric algorithm from the consensus voting logic.

Technical Commentary on Interactive Dynamics:

By manipulating the dashboard, three structural architectural features emerge:

1. **The Drag-Curve Effect (Variable m):** Increasing the value of m compresses the logarithmic curve sharply to the right. Setting a weak Min-Entropy ($h_{min} = 2.0$), an increase from m = 10 to m = 50 shifts the onset of risk emergence (the point at which the probability deviates from 10^{-10}) from 30% collusion to over 80%. This demonstrates the effectiveness of relational decoupling in real time.

2. **The Suppression Effect:** As cryptographic uncertainty increases ($h_{min} > 5$), the curve flattens on the horizontal axis. Regardless of how many nodes the attacker manages to control, epistemic isolation inhibits the adversary's computational capacity. The projected risk remains suppressed until very high compromise fractions, under the selected parameters.
3. The Traditional Consensus: Any parameter combination greater than m = 15 and h_{min} = 1.0 keeps the risk curve flattened towards zero (in logarithmic scale) at the intersection with the 33% dashed line (BFT Marker). The dashboard shows the deviation between models based on consensus thresholds and the CNVS model based on critical fragmentation, residual min-entropy, and non-reducible global validation.

Conclusions of Test 7:

The interactive dashboard shows the effects of Theorem 4 (assumption): the network defends itself by exploiting the geometry (m) and the minimum uncertainty (h_{min}).

12. Analysis of Test 8 (Category 2)

Test Name: Test 8: Interactive Projection (Stress Test) of Dynamic Min-Entropy Erosion.

Test Purpose:

Test 8 transposes the stress analysis of Test 2 into an HTML/JS dashboard application. The goal is to show the user the effects of manipulating the initial structural variables (m and h_{max}) in real time on the Global Veto as a function of the progressive loss and epistemic isolation induced by the increase in colluding nodes.

Formal Assumptions (Rules of Engagement):

- **Assumption 1 (Decay Equations):** The algorithm simultaneously computes three curves based on three residual min-entropy (h_{res}) degradation functions:
    - Linear: hmax · (1 - q)
    - Quadratic: $h_{max} \cdot (1 - q)^2$
    - Radical: $h_{max} \cdot (1 - q)^4$
- **Assumption 2 (Recalculated Inference):** For each curve, the inference limit pinf is dynamically updated with each decimal change in q, impacting the composite probability p_{comp} in a cascading manner.

Technical Commentary on Interactive Dynamics:

By manipulating the dashboard sliders, the internal compensation dynamics of the CNVS framework can be instantly seen:

1. **Absorbing Epistemic Collapse:** By reducing hmax or observing the red curve (Radical Erosion), a rapid increase in the probability of system violation is observed at low

percentages of q. However, by increasing the critical fragments parameter (m), the user can counterbalance the evidence of a reduced initial h_{max} by increasing epistemic isolation with critical fragmentation, thus forcing all three curves downward.

2. **Radical Scenario Neutralization:** By bringing the m slider to standard operating values (e.g. m = 60), the red Radical Erosion curve (the worst-case scenario) is forced to intersect the 10^{-5} threshold above 50% physical collusion.
3. **Resilience Under Moderate Erosion:** For the same parameters (m = 60), the Linear and Quadratic curves (representing realistic metadata exfiltration scenarios) remain anchored to the logarithmic lower bound, showing that the security of the CNVS, under explicit assumptions, is maintained beyond approximately 80% network compromise.

Conclusions of Test 8:

The interactive dashboard shows that a serious data-secrecy compromise does not automatically imply compromise of the entire protocol. The architecture has relational redundancy that can reduce reconstruction probability under explicit assumptions.

13. Analysis of Test 9 (Category 4)

Test Name: Test 9 - Structural Proof-of-Concept (Decoupled Pedagogical Validation Model).

Test Purpose:

Test 9 transposes the theoretical postulates of CNVS into a real application execution environment (Python). The goal is not to measure stochastic projections, but to test the underlying logical infrastructure: demonstrating the implementation of Semantic Binding (opaque selectors), the decoupling between Authentication (Identity) and Truth (Data), and the cascading operation of the three Global Veto Walls (V_G).

Formal Assumptions (Rules of Engagement):

- **Assumption 1 (Pedagogical Abstraction of the Invariant):** For demonstration purposes, the complex polynomial family of geometric invariants (C_{int}) is here replaced by a single linear placeholder equation (c_1).
- **Assumption 2 (Intentional V_L Weakness):** The local barrier (the peripheral node) is assumed to only validate the syntactic formatting of the data (e.g., verify that it is an integer), without any knowledge of its global truthfulness.
- **Assumption 3 (Ordinary Threat Model - Scenarios 2, 3, 4):** The attacker is assumed to know the category of invariants being made public (C_{pub}) and to be able to bypass authentication by stealing the cryptographic identity signatures of legitimate nodes. Epistemic isolation only hides the relational keys and exact parameters (C_{int}).

Terminal Output (Execution Log):

```
--- CNVS EXECUTION ENVIRONMENT INITIALIZATION ---

[Adversary Public View]
{
  "epoch": "time_zero",
  "public_selectors": [
    "tau_f803a31bccff",
    "tau_f430a216296e",
    "tau_f39aebd9412f",
    "tau_ffa9f6f7b0b1"
  ],
  "C_pub": {
    "name": "structural_semantic_consistency",
    "description": "Public category only: the global state is checked through a finite family C={c_i} of structural-semantic consistency constraints. The adversary may know this category
  },
  "not_granted": [
    "theta_C",
    "R_int",
    "hidden_relation_binding",
    "true_values",
    "semantic_key mapping",
    "hidden salts"
  ]
}

[Scenario 1] Honest nodes submit true values with valid identity proofs.
Result: ACCEPTED: Global state validated.

[Scenario 2] The adversary censors the city fragment.
Result: VETO: Relational/topological coherence failed.

[Scenario 3] The adversary knows C_pub and steals identity, but not theta_C/R_int/binding.
The attacker submits a well-typed false value: Piano = 1.
Identity passes, V_L passes, but hidden C should veto.
Result: VETO: Hidden invariant family C failed.

[Scenario 4] The adversary submits correct-looking values with a wrong epoch.
Result: VETO: Identity or epoch authentication failed.

[Scenario 5] Extreme worst case: C_int leakage.
This is NOT the ordinary adversarial view.
Here the attacker is artificially granted theta_C, R_int/binding, and valid identities.
The attacker coordinates Piano = 1 and Metratura = 40 to satisfy the toy c_1.
If this passes, it means the demo is honest: leaking C_int breaks this toy invariant.
Result: ACCEPTED: Global state validated.
```

Technical Commentary on the Data:

Note: For consistency with the variables used in the script shown in the image, the expression will be left in Latex in the following discussion.

System initialization displays the Adversary's Public View: the adversary intercepts fragmented packets identified by hash strings (tau_f80...) and knows it faces a structural check, but is blind to the theta_C parameters, the R_int topology, and hidden salts. The simulator runs five sequential scenarios:

1. **Scenario 1 (Honest Execution):** The ACCEPTED: Global state validated output serves as a baseline. It demonstrates that the system is not affected by a conservative veto (false positive): consistent truth flows unimpeded.
2. **Scenario 2 (Censorship / Topological Incompleteness):** The adversary omits a fragment (City). The Cons_R barrier is triggered before even evaluating the data. The Cons_R barrier prevents false acceptance under topological incompleteness. Availability-level denial-of-service remains a separate implementation concern.
3. **Scenario 3 (Semantic Forging - The Core Test):** The attacker injects a fake data item (Plan = 1). This intentionally bypasses the local syntactic barrier (V_L) and possesses the correct identity hash for that selector and epoch. Nevertheless, the data item enters the computation engine and fails the geometric verification. The output VETO: Hidden invariant family C failed is empirical evidence that in CNVS, node authentication does not guarantee data acceptance.

4. **Scenario 4 (Time Asynchrony / Replay):** The attacker injects real data (stolen from a past transaction), but the time validation matches the selector with a different epoch (wrong_epoch). The VETO: Identity or epoch authentication failed output validates the replay attack block.

   Note: This scenario is retained as a pedagogical approximation of current application mechanisms. In the full CNVS model, replay protection does not depend on simple timing control, but on the integral refresh of selectors, salts, bindings, the R_int topology, and the C_int family.

5. **Scenario 5 (Epistemic Collapse / Worst-Case):** In order to demonstrate architectural transparency, the system is inflicted with the exfiltration of all internal parameters of the hidden invariant (C_{int}). The attacker mathematically computes the false data to solve the assumed equation (Floor = 1, Square Footage = 40).
   The ACCEPTED output validates a false result, and the system is hacked.
   Scenario 5 shows how violating the system invariant leads to its structural collapse.

Conclusions of Test 9:

Scenario 5 shows that the resistance observed in Scenarios 1–4 is conditional on the preservation of epistemic isolation, internal invariant secrecy, and hidden topological relations. If C_{int} is fully exfiltrated, the model can collapse.

In the standard scenarios (1-4), the code shows that bypassing local validation and cryptographic authentication is completely insufficient to corrupt the global state of the CNVS.

The CNVS is shown to be a conditional system, in which security is possible only under specific starting conditions.

14. Analysis of Test 10 (Category 4)

Test Name: Test 10: Advanced Execution Environment (Full Structural-Semantic Model).

Test Purpose:

Test 10 is an executable structural test that illustrates an implementation-level moving-target defense mechanism. It integrates all theoretical components into a single application execution environment: the terminal fragment decomposition, the finite-field polynomial evaluation engine, the Moving Target Defense (Topological Refresh), and inference propagation in dependent mass collusion scenarios. The goal is to measure the strength of the Global Veto (V_G) by pushing the network compromise from 10% to 100% (full physical domain).

Formal Assumptions (Rules of Engagement):

The execution environment does not use predictive formulas, but performs the actual cryptographic computations based on the following implementation assumptions:

- **Assumption 1 (Non-Linear Polynomial Engine):** The hidden invariants (C_{int}) are not linear equations, but combinations of linear, quadratic, cubic and pairwise interactions terms computed in modular arithmetic (Prime Modulus = 1,000,003).
- **Assumption 2 (Integral Topological Refresh):** The system does not defend itself against replay attacks by using a clock controller (timestamp/ epoch). For each transaction, it completely regenerates the selectors, salts, relational links, and the polynomial equation itself.
- **Assumption 3 (Ordinary vs. Upper-Bound Model Separation):** The test runs two parallel simulations. The first simulation respects the Ordinary Threat Model: C_{pub} is known, while C_{int}, hidden binding, and internal relational parameters remain outside the adversarial view. Axiom III is expressed by the non-reducibility of global validation and by the Global Veto. The second simulation programmatically forces the full exfiltration of C_{int} as the simulator's Control Test (Upper-Bound).

Terminal Output (Execution Log):

```
================ SCENARIO 4: FULL REFRESH ================

Same declaration/payload, different CNVS internal structure.
C_int fingerprint at t:     587838ed68d9331b
C_int fingerprint at t + 1: 20b65d29701cf849
h_crit at t:                14
h_crit at t + 1:            15
hidden topology edges at t: 30
hidden topology edges t+1:  35

Replay of old evidence against refreshed instance:
accepted_by_VG: False

Interpretation:
- The previous instance cannot simply be replayed.
- The declaration may be the same, but C_int, binding, topology and selectors changed.
- This models topological and semantic refresh, not a trivial wrong-time rejection.

================ PROGRESSIVE MASS COLLUSION — ORDINARY MODEL, C_pub ONLY ================

collusion | avg_h | avg_ctrl | max_ctrl | P(all critical) | P(false accept) | P(C_int leak break)
-----------------------------------------------------------------------------------------------
     10% | 13.76 |     1.37 |        6 |      0.00000000 |      0.00000000 |          0.00000000
     20% | 13.79 |     2.92 |        9 |      0.00000000 |      0.00000000 |          0.00000000
     30% | 13.78 |     4.21 |       11 |      0.00000000 |      0.00000000 |          0.00000000
     40% | 13.80 |     5.76 |       12 |      0.00000000 |      0.00000000 |          0.00000000
     50% | 13.75 |     7.02 |       13 |      0.00000000 |      0.00000000 |          0.00000000
     60% | 13.76 |     8.32 |       14 |      0.00020000 |      0.00000000 |          0.00000000
     65% | 13.78 |     9.15 |       15 |      0.00190000 |      0.00000000 |          0.00000000
     70% | 13.76 |     9.75 |       15 |      0.00470000 |      0.00000000 |          0.00000000
     75% | 13.78 |    10.40 |       15 |      0.01500000 |      0.00000000 |          0.00000000
     80% | 13.79 |    11.05 |       15 |      0.03340000 |      0.00000000 |          0.00000000
     85% | 13.77 |    11.67 |       15 |      0.07560000 |      0.00000000 |          0.00000000
     90% | 13.77 |    12.52 |       15 |      0.23290000 |      0.00000000 |          0.00000000
     95% | 13.77 |    13.14 |       15 |      0.48640000 |      0.00000000 |          0.00000000
     98% | 13.77 |    13.56 |       15 |      0.79470000 |      0.00000000 |          0.00000000
     99% | 13.80 |    13.58 |       15 |      0.78770000 |      0.00000000 |          0.00000000
    100% | 13.79 |    13.79 |       15 |      1.00000000 |      0.00000000 |          0.00000000

Legend:
- avg_h: average critical threshold h_crit across refreshed instances.
- avg_ctrl: average number of controlled critical fragments.
- P(all critical): threshold breach probability.
- P(false accept): accepted false global state with ordinary C_pub-only view.
- P(C_int leak break): explicit extreme break condition with C_int leakage.

================ PROGRESSIVE MASS COLLUSION — EXTREME C_int LEAK UPPER BOUND ================

collusion | avg_h | avg_ctrl | max_ctrl | P(all critical) | P(false accept) | P(C_int leak break)
-----------------------------------------------------------------------------------------------
     10% | 13.77 |     1.38 |        6 |      0.00000000 |      0.00000000 |          0.00000000
     20% | 13.78 |     2.93 |       10 |      0.00000000 |      0.00000000 |          0.00000000
     30% | 13.80 |     4.24 |       11 |      0.00000000 |      0.00000000 |          0.00000000
     40% | 13.76 |     5.72 |       12 |      0.00000000 |      0.00000000 |          0.00000000
     50% | 13.77 |     7.02 |       13 |      0.00000000 |      0.00000000 |          0.00000000
     60% | 13.77 |     8.30 |       15 |      0.00030000 |      0.00000000 |          0.00030000
     65% | 13.77 |     9.15 |       15 |      0.00190000 |      0.00000000 |          0.00190000
     70% | 13.75 |     9.76 |       15 |      0.00590000 |      0.00000000 |          0.00590000
     75% | 13.79 |    10.41 |       15 |      0.01540000 |      0.00000000 |          0.01540000
     80% | 13.78 |    11.04 |       15 |      0.03740000 |      0.00000000 |          0.03740000
     85% | 13.77 |    11.65 |       15 |      0.07660000 |      0.00000000 |          0.07660000
     90% | 13.78 |    12.51 |       15 |      0.22590000 |      0.00000000 |          0.22590000
     95% | 13.78 |    13.15 |       15 |      0.48930000 |      0.00000000 |          0.48930000
     98% | 13.77 |    13.55 |       15 |      0.78290000 |      0.00000000 |          0.78290000
     99% | 13.80 |    13.60 |       15 |      0.80120000 |      0.00000000 |          0.80120000
    100% | 13.80 |    13.80 |       15 |      1.00000000 |      0.00000000 |          1.00000000

Legend:
- avg_h: average critical threshold h_crit across refreshed instances.
- avg_ctrl: average number of controlled critical fragments.
- P(all critical): threshold breach probability.
- P(false accept): accepted false global state with ordinary C_pub-only view.
- P(C_int leak break): explicit extreme break condition with C_int leakage.

================ FINAL INTERPRETATION ================

1. The corrected Scenario 4 is full refresh, not wrong epoch.
2. Each new instance refreshes topology, binding and invariant family C={c_i}.
3. Progressive mass collusion is tested from 10% up to 100%.
4. P(all critical) measures threshold breach, not automatic false acceptance.
5. P(false accept) measures actual false global acceptance under C_pub-only view.
6. The extreme C_int leak run is a deliberately pessimistic upper-bound break model.
```

Technical Commentary on the Data:

The output is divided into two distinct evidentiary sections.

1. Topological Refresh Dynamics (Scenario 4):

- Given the same declarative payload sent at times t and t + 1, the engine generates two radically different internal fingerprints (Fingerprint C_int: 587838... vs 20b65d...).
- The number of hidden topological vertices also changes (from 30 to 35).
- By injecting the exact evidence (cryptographic hashes) breached at time t into the instance at time t + 1, the outcome is accepted_by_VG: False. The executable test illustrates an implementation-level moving-target defense mechanism, showing that evidence captured at time t is rejected when replayed against the refreshed instance at time t + 1.

2. Mass Collusion Matrix (Ordinary vs. Exfiltration Model):

- **P(all critical) column:** Measures the probability that the adversary will gain physical control of all the fragments needed for the attack. According to statistical laws, this probability increases as q increases, reaching 1.00000000 at 100% collusion.
- **Ordinary Model (P(false accept)):** Under the ordinary model implemented in the simulator, even full compromise of peripheral verifiers does not by itself produce observed false acceptance, provided C_int remains outside the adversarial view. This indicates that, in the tested model, peripheral verifier compromise alone is insufficient for global falsification without access to the hidden invariant structure.
- **Upper-Bound Model (P(C_int leak break)):** In the second table, programmatic exfiltration of the C_int invariant allows the attacker to compute the unknowns. Consequently, the system failure rate increases consistently with the node control probability, showing a progressive increase in the system failure probability starting from 60% collusion percentages and reaching the system failure certainty (1.00000000) when node control is 100%. This data is crucial for due diligence: it demonstrates that the results in the previous table are not a software artifact, but the natural cryptographic consequence of the principles on which the CNVS logic is based.

Test 10 Conclusions:

Test 10 demonstrates the operational behavior of the CNVS framework, under the explicit assumptions of the Theory (secrecy of C_int), but without assuming its formulas as an implementation condition of the script. By running the full Structural-Semantic model, the executable model illustrates how state validation can be based on hidden geometric invariants under the stated implementation assumptions. In the simulated ordinary model, even when

100% of the peripheral verifiers are compromised, global falsification is not observed as long as C_int remains excluded from the adversarial view.

Analysis of Test 11 (Category 4)

Test Name: Test 11 - Executable Fragmentation Sensitivity under Hidden Invariant Binding

Test Purpose:

Test 11 extends the executable structural model to measure the sensitivity of the Global Validation (V_G) acceptance probability as the critical fragmentation cardinality (m) increases. The goal is to evaluate, in an executable structural model, whether increasing the critical fragmentation cardinality m reduces the observed unauthorized V_G acceptance probability under severe peripheral compromise (q = 60%, 80%, 90%). Instead of relying on statistical formulas, the test evaluates adversarial state candidates by running them through the executable validation pipeline (V_L -> $Cons_R$ -> Inv_C -> V_G).

Formal Assumptions (Rules of Engagement):

The execution environment does not use predictive statistical formulas to determine adversarial success, but performs the actual cryptographic computations based on the following implementation assumptions:

- **Assumption 1 (Finite-Field Execution Pipeline):** The acceptance of a candidate is decided by executing relational pairwise constraints and algebraic tags in modular arithmetic (Prime Modulus = 1,000,003). Local verifiers only check syntactic admissibility (domain bounds), while the global layer verifies the hidden invariant binding.
- **Assumption 2 (Ordinary Adversarial Model):** The adversary holds a proportion q of the peripheral verifying nodes. Missing critical fragments are inferred with a probability bounded by the residual min-entropy ($p_{inf} \leq 2^{-h\_min}$). Incorrect guesses generate values that are well-formed (passing local syntax) but semantically false.
- **Assumption 3 (Upper-Bound C_int Leak Control):** The test runs a parallel simulation where the hidden invariant (C_{int}) is completely exfiltrated. The adversary uses this leaked data to algebraically invert the equations and reconstruct the missing critical values, functioning as a structural sanity check for the simulator.

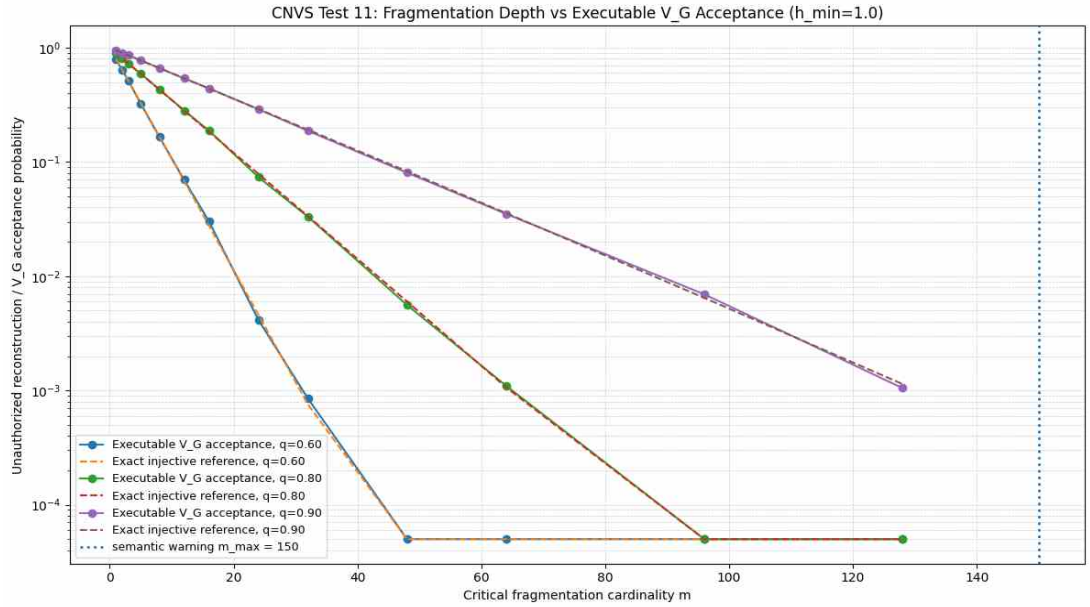
Terminal Output (Execution Log):

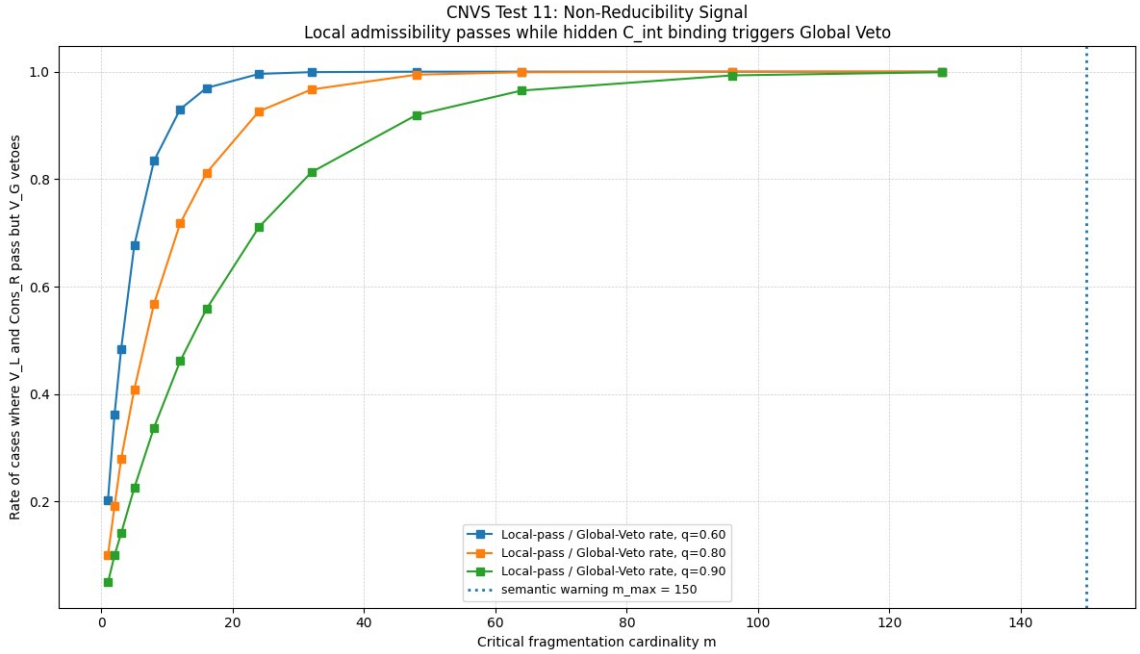
```
CNVS Test 11: Executable Fragmentation Sensitivity Test
------------------------------------------------------------
Q_verifiers = 1000
h_min = 1.0
iterations per point = 20000
m values = [1, 2, 3, 5, 8, 12, 16, 24, 32, 48, 64, 96, 128]
V_G acceptance is computed by executing V_L -> Cons_R -> Inv_C -> V_G.
The analytical formula is shown only as a reference.


=== q = 0.60 ===
m | k | VG_accept | local-pass/VG-veto | exact_ref | theorem_ref | C_int_leak
  1 |  50 | 0.79690000 | 0.20310000 | 0.80000000 | 0.80000000 | True
  2 |  50 | 0.63755000 | 0.36245000 | 0.63993994 | 0.64000000 | True
  3 |  50 | 0.51615000 | 0.48385000 | 0.51185584 | 0.51200000 | True
  5 |  50 | 0.32355000 | 0.67645000 | 0.32737246 | 0.32768000 | True
  8 |  50 | 0.16600000 | 0.83400000 | 0.16733141 | 0.16777216 | True
 12 |  50 | 0.06970000 | 0.93030000 | 0.06829439 | 0.06871948 | True
 16 |  50 | 0.03035000 | 0.96965000 | 0.02783151 | 0.02814750 | True
 24 |  50 | 0.00415000 | 0.99585000 | 0.00460115 | 0.00472237 | True
 32 |  64 | 0.00085000 | 0.99915000 | 0.00075606 | 0.00079228 | True
 48 |  96 | 0.00005000 | 0.99995000 | 0.00002004 | 0.00002230 | True
 64 | 128 | 0.00000000 | 1.00000000 | 0.00000052 | 0.00000063 | True
 96 | 192 | 0.00000000 | 1.00000000 | 0.00000000 | 0.00000000 | True
128 | 256 | 0.00000000 | 1.00000000 | 0.00000000 | 0.00000000 | True

=== q = 0.80 ===
m | k | VG_accept | local-pass/VG-veto | exact_ref | theorem_ref | C_int_leak
  1 |  50 | 0.89975000 | 0.10025000 | 0.90000000 | 0.90000000 | True
  2 |  50 | 0.80815000 | 0.19185000 | 0.80995996 | 0.81000000 | True
  3 |  50 | 0.72080000 | 0.27920000 | 0.72889187 | 0.72900000 | True
  5 |  50 | 0.59150000 | 0.40850000 | 0.59019793 | 0.59049000 | True
  8 |  50 | 0.43230000 | 0.56770000 | 0.42987082 | 0.43046721 | True
 12 |  50 | 0.28210000 | 0.71790000 | 0.28150708 | 0.28242954 | True
 16 |  50 | 0.18840000 | 0.81160000 | 0.18420188 | 0.18530202 | True
 24 |  50 | 0.07365000 | 0.92635000 | 0.07867907 | 0.07976644 | True
 32 |  64 | 0.03315000 | 0.96685000 | 0.03349842 | 0.03433684 | True
 48 |  96 | 0.00560000 | 0.99440000 | 0.00601336 | 0.00636269 | True
 64 | 128 | 0.00110000 | 0.99890000 | 0.00106536 | 0.00117902 | True
 96 | 192 | 0.00005000 | 0.99995000 | 0.00003212 | 0.00004048 | True
128 | 256 | 0.00000000 | 1.00000000 | 0.00000092 | 0.00000139 | True

=== q = 0.90 ===
m | k | VG_accept | local-pass/VG-veto | exact_ref | theorem_ref | C_int_leak
  1 |  50 | 0.95015000 | 0.04985000 | 0.95000000 | 0.95000000 | True
  2 |  50 | 0.89985000 | 0.10015000 | 0.90247748 | 0.90250000 | True
  3 |  50 | 0.85930000 | 0.14070000 | 0.85731079 | 0.85737500 | True
  5 |  50 | 0.77465000 | 0.22535000 | 0.77358768 | 0.77378094 | True
  8 |  50 | 0.66255000 | 0.33745000 | 0.66295616 | 0.66342043 | True
 12 |  50 | 0.53850000 | 0.46150000 | 0.53946807 | 0.54036009 | True
 16 |  50 | 0.44165000 | 0.55835000 | 0.43880493 | 0.44012667 | True
 24 |  50 | 0.28905000 | 0.71095000 | 0.28997121 | 0.29198902 | True
 32 |  64 | 0.18705000 | 0.81295000 | 0.19130651 | 0.19371148 | True
 48 |  96 | 0.08045000 | 0.91955000 | 0.08285768 | 0.08525759 | True
 64 | 128 | 0.03515000 | 0.96485000 | 0.03564820 | 0.03752414 | True
 96 | 192 | 0.00695000 | 0.99305000 | 0.00646513 | 0.00726886 | True
128 | 256 | 0.00105000 | 0.99895000 | 0.00114021 | 0.00140806 | True
```





Technical Commentary on the Data:

The output is divided into three distinct evidentiary sections.

Algorithmic Convergence to Theoretical Bounds:

The executable V_G acceptance rate closely follows the exact injective-assignment reference within Monte Carlo resolution.

As the critical fragmentation cardinality (m) expands, the probability of an unauthorized state reconstruction monotonically decreases. At q = 0.90 (90% network compromise) and m = 128, the empirical acceptance rate collapses to 0.00105000, which closely aligns with the expected exact theoretical reference of 0.00114021. This supports the executable consistency of the structural model with the expected fragmentation-driven decay.

The Non-Reducibility Signal (Global Veto Dynamics):

The "local-pass/VG-veto" column measures the structural decoupling of local syntax and global semantics.

The data demonstrates that as m increases, the rate at which an adversary successfully bypasses local syntactic checks (V_L) and relational consistency ($Cons_R$) approaches 1.0. Specifically, at q = 0.90 and m = 128, 99.895% of forged states are deemed valid by peripheral nodes, yet they trigger a Global Veto under the hidden invariant-binding checks.

Epistemic Isolation Integrity (C_{int} Leak Control):

The "C_int_leak" column returns True uniformly across all evaluated dimensions.

This programmatic exfiltration verifies that if the hidden cryptographic parameters are exposed, the execution engine deterministically fails and accepts the forged state. This control data is fundamental for technical due diligence: it supports the conclusion that, in the ordinary model, the observed resistance depends on preservation of the hidden invariant structure rather than on a hard-coded rejection artifact.

Test 11 Conclusions:

Test 11 provides executable support for the role of critical fragmentation inside the CNVS framework. Under the ordinary model, where C_{int} remains outside the adversarial view, increasing the critical fragmentation cardinality m progressively reduces the observed probability of unauthorized V_G acceptance, even under high peripheral verifier compromise.

The local-pass / V_G-veto curve shows that this reduction is not caused by local syntactic rejection. In most failed adversarial attempts, V_L and $Cons_R$ are satisfied, while V_G rejects the candidate because the hidden invariant binding Inv_C is not satisfied. This illustrates the CNVS non-reducibility principle: local admissibility does not imply global validity.

The C_{int} leak control confirms the conditional nature of the model.
When the hidden invariant structure is programmatically exfiltrated, the adversary can reconstruct the missing critical values and force acceptance. Therefore, Test 11 does not establish unconditional security, but strengthens the operational plausibility of CNVS under its stated assumptions.

In relation to the previous tests, Test 11 bridges Test 5 and Test 10: Test 5 provides the provisioning formula for m_{min}, Test 10 demonstrates executable separation between

peripheral compromise and global falsification, and Test 11 shows how increasing m affects V_G acceptance inside the executable structural model itself.

16. General Conclusions of the MTC Report

The Monte Carlo Test Infrastructure (MTC) and Application Execution Environments subjected the Closed Native Verification Systems (CNVS) framework to a cryptographic, stochastic, and structural stress test.
The projections and executable evidence collected from the 11 documented tests showed mutually consistent behavior under the stated assumptions.
The data analysis led to four architectural conclusions:

1. Conditional Deviation from Majority-Based Validation

Many classical distributed validation paradigms rely on quorum, majority, or consensus thresholds. CNVS is evaluated here under a different threat model, where global state acceptance is not reducible to the fraction of honest nodes alone.
The numerical projection tests (Tests 1 and 2) and the executable structural model (Test 10) support the conclusion that the CNVS architecture severs this link.
Assuming Epistemic Isolation is maintained, the framework suppresses observed falsification probability even at very high peripheral-compromise rates under the tested conditions, provided C_int remains outside the adversarial view.
Security does not derive solely from majority-based data replication, but also from critical fragmentation, hidden binding, and non-reducible global validation.

2. Numerical Consistency of the Design Equation

The equation $\mathbb{P}(\text{Rec}^*) \leq [q + (1 - q)\, 2^{-h\_min}]^m$ predicts network behavior under attack. Dynamic erosion tests (Tests 2, 8) and asymmetric topological exposure tests (Test 5) show that increasing the semantic granularity (m) can reduce reconstruction probability under the selected erosion and exposure assumptions. The system provides a formula (Eq. 42) to mathematically size the infrastructure based on precise risk targets.

3. Temporal Refresh and Moving-Target Mitigation

The integration of Topological Refresh and limited validation windows (Test 4) transforms the network from a static target to a reactive ecosystem. Fault-injection and replay attempts are mitigated in the simulated model by the continuous mutation of polynomial parameters (C_{int}), but also induce a Sybil Purge that actively degrades the adversary's offensive capability with each failed cycle.

4. Asymmetric Crypto-Economic Security (Truth-Forcing)

The combination of Global Veto (V_G) and economic penalty (Slashing) creates a parametric *truth-forcing model* (Tests 3, 6). Controlling 80% of the network in a CNVS ecosystem does not guarantee state control, but it exposes the adversary to a profoundly negative Expected Value.

Infiltration attempts may become financially counterproductive under the selected slashing assumptions.

Final Verdict:

The MTC suite provides consistent numerical and applicative support for the behavior predicted by the CNVS framework under the stated formal assumptions.
The tests do not replace the formal proof of the Technical Document and do not establish unconditional security.
Rather, they show that, when the epistemic isolation of C_int is preserved, compromise of peripheral verifiers does not by itself imply falsification of the global state.

The projective tests confirm the numerical consistency of the relationship between adversarial fraction q, critical cardinality m, and residual min-entropy h_{min}. The interactive tests make this relationship explorable in real time. The executable structural tests demonstrate the separation between local validation, identity authentication, and global acceptance, highlighting the role of the Global Veto, hidden invariant binding, and topological/invariant refresh.

Within the ordinary CNVS model, security does not derive from a simple majority of honest nodes, but from the non-reducibility of global validation and the isolation of the internal structure $C_{int} = <\theta_C, R_{int}, \text{binding}>$.

AI Assistance and Authorship Declaration

The source code and executable simulation scripts included in this suite were developed with the assistive capabilities of Artificial Intelligence (Large Language Models). The author explicitly declares that all AI-generated outputs have been rigorously reviewed, manually corrected, validated, and stress-tested to ensure strict cryptographic and mathematical adherence to the underlying theoretical framework. The core implementation logic, the architecture of the testing environment, the structural design, and the overall organization of the suite are entirely the original intellectual creation of the human author, who assumes full and exclusive responsibility for the scientific integrity and accuracy of the final code.

Author:
Massimo Comitato
Italy, Milano (MI),
08 - 07 - 2026