

Governance Stability Simulator

A Control-Theoretic Model of Institutional Adaptation

An open analytical framework modelling governance as a feedback control system, demonstrating how latency and signal fidelity determine the structural stability limits of any institutional architecture.

Björn Kenneth Holmström

February 2026

Creative Commons Attribution-ShareAlike 4.0 International

<https://bjorkkennethholmstrom.org/working-papers/governance-stability-simulator>

Executive summary

Governance systems fail in predictable ways. Not because leaders lack wisdom or institutions lack resources, but because the underlying architecture generates failure as a structural output. This is not a political claim. It is an engineering observation.

This paper introduces the **Governance Stability Simulator** — an open analytical framework that models governance as a feedback control system. Using standard mathematics from control theory and cybernetics, it becomes possible to compare institutional architectures not by their stated intentions, but by their measurable stability properties: how quickly they recover from shocks, how accurately they perceive the systems they govern, and whether their response mechanisms are structurally capable of matching the complexity they face.

The simulator demonstrates that three parameters — **latency** (the delay between a crisis and a policy response), **signal fidelity** (the accuracy of information reaching decision-makers), and **controller gain** (the aggressiveness of the intervention) — interact in ways that place hard mathematical ceilings on what any governance architecture can achieve. These ceilings are not negotiable through better intentions or increased funding. They are topological constraints.

The framework is presented as an analytical tool, not a political prescription. It does not advocate for specific policies or institutional arrangements. It provides a formal language for comparing governance architectures the way engineers compare control systems — by their demonstrated performance under defined conditions.

All code is open source and available for inspection, replication, and extension.

Part I: Governance as a feedback system

The engineering analogy is not a metaphor

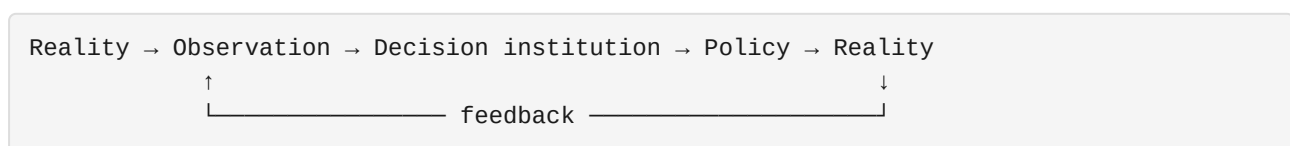
When engineers design systems that must maintain stability under external disturbance — aircraft, power grids, chemical plants — they use a formal discipline called control theory. The discipline provides precise methods for analyzing whether a system will remain stable, how quickly it will recover from shocks, and what design constraints limit its performance.

Governance systems contain every structural element that control theory was developed to analyze. They receive information about the state of the world they govern. They process that information through institutions. They produce interventions intended to correct deviations from desired conditions. And their outputs feed back into the world, producing new states that must be observed and acted upon again.

This is not analogy. It is structural identity.

Control theory	Governance equivalent
System state $x(t)$	Societal condition (wellbeing, stability, resource levels)
Sensors / observations	Economic indicators, local reporting, citizen feedback
Controller	Decision-making institutions
Actuators	Policy interventions, resource allocation
Disturbance $d(t)$	Crises, shocks, external disruptions
Latency τ	Time from crisis to implemented policy response
Signal noise σ	Information distortion, aggregation loss, measurement error
Feedback loop	Institutional adaptation based on observed outcomes

The feedback structure of any governance system can be drawn as follows:



Every element in this diagram has a governance equivalent. And every element can fail in ways that produce predictable instability.

Why this matters: the visibility problem

The most consequential insight from control theory is that system performance is determined not just by the quality of decisions, but by the quality of the information on which those decisions are made — and by the delay between when a problem emerges and when a corrective action takes effect.

A perfectly competent institution operating on corrupted or delayed information will produce systematically worse outcomes than a less sophisticated institution with accurate, timely signals. This is not a failure of competence. It is a failure of observability — the formal term for whether a system's true state can be reconstructed from available measurements.

Many governance failures that appear to be failures of political will or institutional competence are, on inspection, failures of observability and latency. The institution is responding to the world it can see, not the world that exists. And by the time its response arrives, the world has moved on.

The historical context

Control theory emerged as a formal discipline in the mid-twentieth century, developed by mathematicians and engineers including Norbert Wiener, whose 1948 work

Cybernetics

explicitly extended its principles to social and biological systems. The parallel development of cybernetics — the science of feedback in complex systems — produced thinkers like Ross Ashby, whose Law of Requisite Variety provides one of the foundational theorems applied in this framework, and Stafford Beer, who spent decades attempting to apply these principles to organizational and national governance.

These efforts largely stalled — not because the concepts were wrong, but because the computational and communicative infrastructure needed to implement them did not yet exist. The theoretical work remained ahead of the practical tools.

The governance simulator presented here applies these same principles using contemporary computational methods. The mathematics is not new. The application is.

Part II: A formal grammar for governance

Seven primitives

Any governance system — from a municipal council to a continental federation — can be represented using seven structural primitives. Together these constitute a minimal formal grammar sufficient to model, compare, and analyze institutional architectures.

1. Nodes

A node is any entity capable of receiving information, processing it, and producing an action. Nodes exist at every scale: an individual citizen, a local authority, a national ministry, an international body. The critical property of a node is its **processing capacity** — the complexity of signals it can interpret and respond to meaningfully.

Ashby's Law of Requisite Variety states that a controller must possess at least as much variety (complexity) as the system it seeks to govern. A node whose processing capacity is smaller than the complexity of its domain cannot govern that domain stably, regardless of its formal authority.

2. State

The state $x(t)$ is the condition of a node or system at time t . It is what is actually true about the world — the real level of wellbeing, stability, or resource availability in a community. State variables change over time in response to disturbances and interventions.

Formally:

$$x(t+1) = A \cdot x(t) + B \cdot u(t-\tau) + d(t)$$

Where A captures natural dynamics (decay, growth), B captures the effectiveness of interventions u , τ is latency, and $d(t)$ represents external disturbances. The distinction between state and observation is foundational: governance systems act on what they *observe*, which may differ significantly from what is *true*.

3. Flows

Flows are the movement of information or resources between nodes. An information flow carries signals about the state of the world. A resource flow carries interventions — funding, personnel, policy mandates. The structure of flows determines which nodes can perceive which parts of the system, and which nodes can

act on which parts.

Flow architecture is a primary determinant of governance performance. A system in which all information must pass through a single central node before action can be taken has fundamentally different stability properties than one in which nodes communicate laterally and act locally.

4. Latency

Latency τ is the dead-time between a signal entering the system and a corrective action reaching the affected node. In governance systems, latency accumulates across multiple stages: detection, reporting, aggregation, deliberation, decision, legislation, implementation.

Latency has a precise and important consequence: it places a hard ceiling on the control gain K that a stable system can use. The relationship is approximately:

$$K_{\max} \approx 1 / (\tau \cdot |A|)$$

This means that a governance system with high latency is

structurally incapable

of responding aggressively to crises, regardless of political will. Attempting to increase responsiveness beyond this ceiling produces oscillation and instability. This constraint is mathematical, not political.

5. Constraints

Constraints are hard limits that the system cannot safely cross. In physical systems these include actuator limits, material stress thresholds, and conservation laws. In governance systems they include ecological boundaries (which cannot be exceeded without systemic damage), minimum dignity thresholds (below which social cohesion breaks down), and coordination requirements (which cannot be abandoned without losing system-wide function).

Constraints define the feasible operating space. A governance architecture that routinely operates near constraint boundaries is structurally fragile; one that maintains comfortable margins is robust.

6. Feedback loops

A feedback loop is the mechanism by which the outcomes of governance actions return to influence future decisions. Negative feedback loops are stabilizing — they correct deviations from a target state. Positive feedback loops are destabilizing — they amplify deviations.

The quality of a feedback loop depends on two things: its speed (how quickly outcomes are observed and acted upon) and its accuracy (whether the observed signal faithfully represents the true state). A slow or inaccurate feedback loop is worse than no feedback loop, because it produces interventions calibrated to a

reality that no longer exists.

7. Signal fidelity

Signal fidelity is the accuracy of information as it moves through the system. Every measurement introduces noise. Every aggregation discards information. Every layer of reporting introduces potential for distortion, selective emphasis, or motivated misrepresentation.

Formally, the observed state $y(t)$ differs from the true state $x(t)$:

$$y(t) = x(t) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$$

High signal fidelity means σ is small — the controller acts on information close to reality. Low signal fidelity means σ is large — the controller responds to a corrupted image of the world. The consequences compound with latency: a system that observes inaccurately *and* acts slowly is doubly handicapped, because by the time a distorted signal produces a delayed response, the underlying reality may have changed entirely.

The two fundamental failure modes

These seven primitives generate two structural failure modes that recur across governance contexts at every scale.

The observability failure occurs when signal fidelity is insufficient for the controller to reconstruct the true state of the system. The controller makes decisions based on a systematically distorted picture of reality. Interventions are miscalibrated not because of poor judgment, but because the available information does not support better judgment. No amount of institutional competence compensates for this failure — it is architectural.

The latency-gain trap occurs when high latency forces the system into a low-gain regime. The controller can only respond weakly to detected deviations, because stronger responses would cause oscillation. The system drifts persistently away from target states not because it is unresponsive, but because its response is structurally capped below the level needed to match the speed of external disturbances.

Both failures are diagnosable in advance from the structure of the governance architecture. And both are addressable through architectural changes — specifically, changes that reduce latency and improve signal fidelity at the point where decisions are made.

Part III: The simulation

Scenario design

The simulator models a network of ten coupled nodes — representing any collection of governance units at the same scale: municipalities, regions, provinces, or member states. Each node has a true stability state $x_i(t)$ representing its condition at time t , initialized at equilibrium.

At time step 20, a localized shock strikes two nodes (nodes 2 and 7). The remaining eight nodes are undisturbed. This is the canonical scenario for testing subsidiarity: a crisis that is real and severe at specific locations, but absent elsewhere.

Two governance architectures are then compared under identical shock conditions.

Architecture A: centralized control

In Architecture A, all ten nodes report upward to a central controller. The controller observes a **national mean** — the average condition across all ten nodes — and applies a uniform intervention to the entire network.

The structural consequences follow directly from the primitives:

- **Latency** $\tau_A = 12$: the signal must travel up through reporting layers, be processed centrally, and a policy response must travel back down and be implemented. Twelve time steps of dead-time.
- **Signal noise** $\sigma_A = 6.0$: aggregating ten local signals into a national mean destroys spatial information. The central controller cannot distinguish a severe local crisis from a mild system-wide fluctuation. A shock of magnitude -45 at two nodes appears, from the center, as a modest dip in the national average.
- **Gain ceiling** $K_A = 0.30$: with latency of 12, the stability ceiling constrains the controller to weak responses. Attempting to increase gain beyond this causes oscillation.

The controller's response is therefore simultaneously under-powered for the crisis nodes and broadcast uniformly across nodes that need no intervention at all.

Architecture B: distributed / fractal control

In Architecture B, each node observes its own condition directly and applies its own corrective intervention. A lateral coordination layer shares information across nodes, but decision authority and response capacity sit locally.

- **Latency** $\tau_B = 2$: local controllers act within days rather than years. The dead-time is the minimum required for local observation and response.
- **Signal noise** $\sigma_B = 0.5$: local controllers observe local conditions with high fidelity. No aggregation loss. The crisis nodes see exactly how severe their situation is.
- **Gain** $K_B = 0.45$: the lower latency permits a stronger response while remaining within the stability ceiling. Note that this is still a constrained value — the ceiling exists in distributed systems too, and ignoring it produces instability regardless of architecture (see the limitations section).

Simulation output

The simulator produces four visualizations from a single run:

Heatmaps (node × time): The most diagnostic output. Architecture A shows the crisis spreading and persisting across the network as the delayed, uniform response fails to contain it and disrupts healthy nodes. Architecture B shows the crisis contained to nodes 2 and 7, with the remaining nodes unaffected throughout.

Node traces: Individual stability trajectories for crisis and healthy nodes. In Architecture A, healthy nodes exhibit significant disruption from the uniform policy — they receive an intervention calibrated to a national mean that includes their stable condition alongside the crisis, producing an over-correction. In Architecture B, healthy node traces are nearly flat throughout the crisis period.

Cumulative deficit bar chart: The integral of stability loss below the equilibrium target, per node, across the full simulation. This captures both the depth and duration of the deficit. Architecture A produces substantial deficits at non-crisis nodes as collateral damage from the uniform response. Architecture B concentrates deficit at the crisis nodes, with minimal collateral impact elsewhere.

Control signal (crisis node): Architecture A's controller responds to the diluted national mean — a weak signal that substantially underestimates the local severity. Architecture B's controller responds to the local state directly, applying a proportionate intervention immediately.

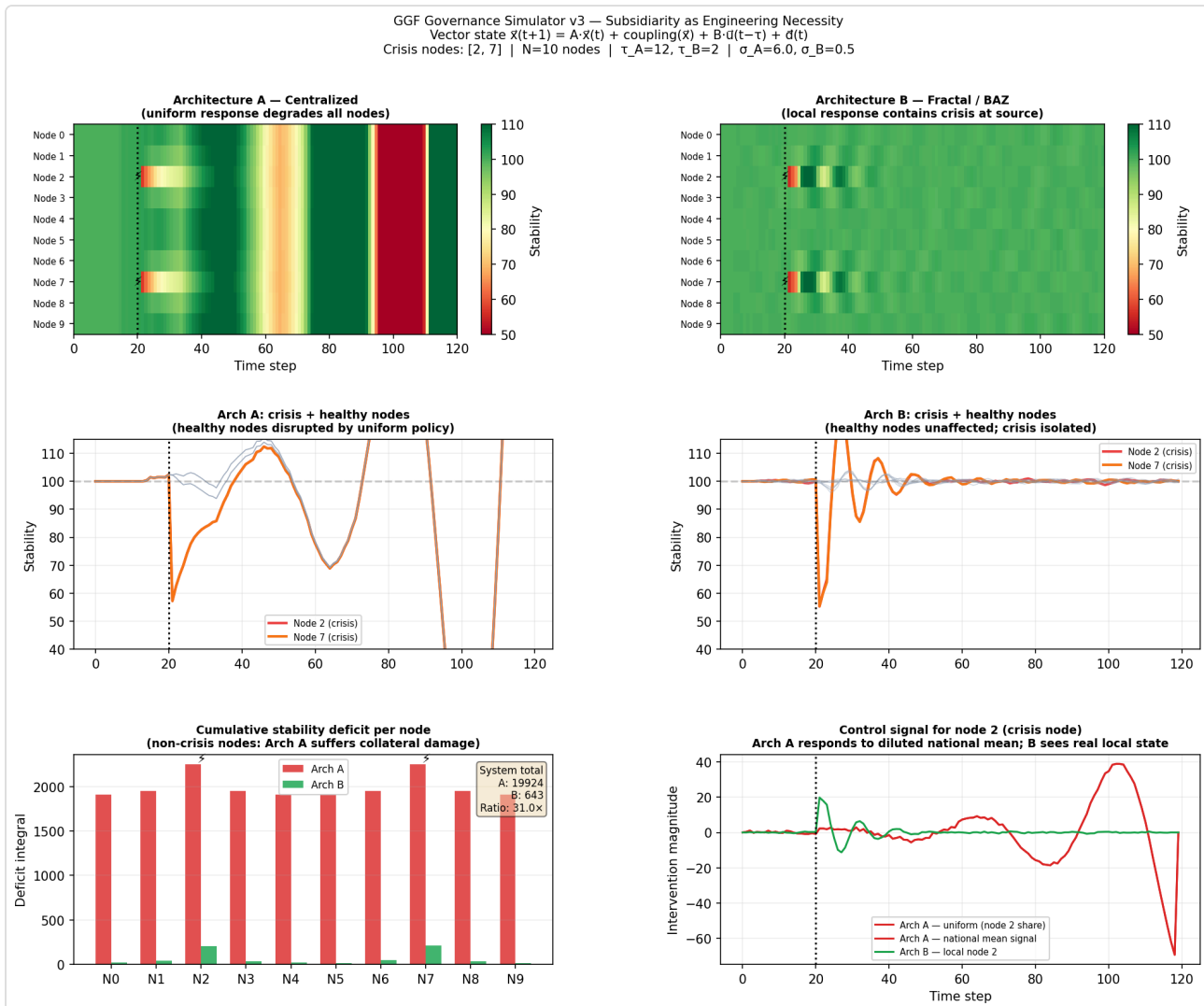


Figure 1: GGF Governance Simulator v3 output. Top row: stability heatmaps for Architecture A (centralized) and Architecture B (fractal/distributed), showing node conditions over 120 time steps. Crisis nodes 2 and 7 are marked. Middle row: individual node traces, showing collateral disruption to healthy nodes under Architecture A and isolation of the crisis under Architecture B. Bottom left: cumulative stability deficit per node. Bottom right: control signal for crisis node 2, showing Architecture A responding to a diluted national mean while Architecture B responds to the true local state.

The averaging problem

The central structural finding is what might be called the averaging problem. When a centralized controller aggregates local signals into a single mean, two things happen simultaneously:

First, the severity of localized crises is systematically underestimated. A shock of -45 at two of ten nodes appears as a deviation of approximately -9 from the national mean. The controller responds to the -9 , not the -45 .

Second, the uniform response applies an intervention sized for -9 across all ten nodes. For the eight healthy nodes, this is an unsolicited disruption. For the two crisis nodes, it is an intervention five times weaker than the actual disturbance requires.

The averaging problem is not a failure of the central controller's competence or resources. It is a consequence of the architecture. Spatial information —

where

the problem is — is destroyed by aggregation. No improvement to the quality of central decision-making recovers that lost information, because the information was discarded before it arrived.

Subsidiarity — the principle that decisions should be made at the lowest level capable of handling them — is, in control-theoretic terms, the prescription that follows directly from the averaging problem. It is an engineering requirement before it is a political preference.

Part IV: Structural observations

The simulation produces several observations that hold across parameter variations and are grounded in established control theory. They are presented here as structural findings, not policy conclusions.

Latency is the primary determinant of maximum responsiveness

The relationship between latency and the gain ceiling is the most consequential finding for governance design. It means that the speed of a governance system's response is not primarily a function of political will, institutional quality, or available resources. It is a function of the time required for information to travel from where a problem exists to where a decision is made, and for a response to travel back.

This places a hard limit on what centralized governance can achieve in high-latency environments, regardless of its other qualities. A system with twelve time steps of latency cannot match the crisis response of a system with two time steps of latency, even if every other parameter is identical. The physics of feedback do not make exceptions for institutional seniority or formal authority.

Signal fidelity determines whether the system is responding to reality

A controller with low signal fidelity is, in a precise sense, governing a fiction — a distorted representation of the world constructed from noisy, aggregated, selectively filtered signals. The interventions it produces are calibrated to that fiction. When the fiction diverges significantly from reality, the interventions are systematically miscalibrated.

Signal fidelity degrades predictably with the distance between where a condition exists and where it is observed. It degrades with each aggregation step that discards local information in favor of summary statistics. It degrades with each reporting layer that introduces motivated distortion or bureaucratic simplification. And it degrades with time: the longer a signal takes to travel, the more the underlying reality may have changed by the time it arrives.

Collateral disruption is structural, not incidental

In the simulation, healthy nodes suffer significant stability deficits under Architecture A despite experiencing no shock themselves. This collateral disruption is not a modelling artifact. It reflects the structural consequence of applying uniform interventions to a heterogeneous system.

Any governance system that responds to averaged signals with uniform policies will produce interventions that are simultaneously too weak for the places that need them and too strong for the places that do not. The collateral cost is not a side effect that better calibration can eliminate. It is a direct consequence of the information loss from aggregation.

Coupling amplifies the cost of delayed response

The simulator includes a coupling term that models contagion — the tendency of instability at one node to propagate to neighboring nodes over time. Under Architecture A's longer latency, the crisis at nodes 2 and 7 has time to bleed into adjacent nodes before the response arrives. Under Architecture B's shorter latency, the crisis is contained before contagion has time to develop.

This means the performance gap between architectures is not fixed — it grows with crisis severity and duration. The longer a response takes, the larger the network that becomes affected, and the more difficult the recovery problem becomes. High-latency architectures face compounding costs that low-latency architectures avoid entirely.

The distributed gain ceiling is real

A finding that deserves explicit emphasis: Architecture B is not immune to stability constraints. Distributed systems with too-aggressive local controllers will oscillate and destabilize, as demonstrated during the development of this simulator. The gain ceiling applies to every feedback system regardless of its topology.

What changes under distributed architecture is not the existence of the ceiling, but its height. Lower latency permits a higher ceiling, which permits more aggressive responses. But the ceiling must still be respected. This has an important governance implication: local autonomy without coordination protocols can produce its own instability. The benefit of distributed architecture is only realized when local controllers operate within bounds established by a shared coordination layer — which is precisely the role of protocol-level governance as distinct from directive governance.

Performance differences are quantifiable

The simulation produces objective performance metrics: recovery time per node, cumulative stability deficit, and collateral deficit at non-crisis nodes. These are not rhetorical claims. They are numbers produced by running the model under specified parameters.

This quantifiability is the key property that distinguishes the engineering framing from the political framing. It becomes possible to ask not "which architecture is better in principle" but "what is the measured performance difference under these conditions, and how does it change as parameters vary." The answer will

depend on the specific parameters chosen — which is why the limitations section addresses parameter selection carefully.

Part V: Limitations

A simulation that does not state its limitations is an argument in disguise. The following limitations are inherent to the current model and should inform how its findings are interpreted and applied.

The parameters are illustrative, not empirical

The specific values used in the simulation — latency of 12 versus 2, noise of 6.0 versus 0.5, a shock of magnitude 45 — are chosen to produce legible structural contrasts, not to represent measured properties of any real governance system. The qualitative findings (that high latency caps responsiveness, that aggregation destroys spatial information, that coupling amplifies unresolved crises) are robust to parameter variation. The specific quantitative outputs — recovery times, deficit integrals, performance ratios — are artifacts of the chosen parameters and should not be cited as empirical measurements.

Grounding this framework in real governance data would require empirical work: measuring actual latency distributions across governance layers, estimating information loss across reporting hierarchies, and calibrating coupling parameters from historical crisis propagation data. That work is outside the scope of this paper but represents a natural and important extension.

The model is linear

The state transition equation $x(t+1) = A \cdot x(t) + B \cdot u(t-\tau) + d(t)$ is a linear time-invariant model. Real governance systems are nonlinear. Stability thresholds are not smooth — systems often appear stable until they cross a critical point and then fail rapidly. Feedback gains are not fixed — institutions adapt their response strategies over time. The interaction between crisis severity and response capacity is not multiplicative in the simple way the model assumes.

Linear models are the correct starting point: they are analytically tractable, their properties are well understood, and they capture the first-order behavior that dominates in the regime near equilibrium. But governance crises often involve precisely the nonlinear dynamics that linear models cannot represent — cascading failures, tipping points, hysteresis. Extensions to nonlinear dynamics are a significant research direction.

Nodes are treated as homogeneous

In the current model, all ten nodes have the same dynamics, the same processing capacity, and the same coupling strength to their neighbors. Real governance units are heterogeneous in all these dimensions: a dense urban node has different dynamics than a dispersed rural one; a well-resourced municipality has different response capacity than an underfunded one; geographic and economic proximity creates asymmetric coupling.

Heterogeneous network models would produce richer and more realistic dynamics. They would also allow exploration of how inequality in node capacity interacts with governance architecture — a question of significant practical importance.

The model has a single disturbance type

The simulation uses a single instantaneous shock to two nodes. Real governance environments involve continuous, overlapping, correlated disturbances of varying severity and spatial extent. Some crises are truly localized; others are system-wide. Some are sudden; others accumulate slowly. Some are correlated across nodes; others are independent.

The localized shock scenario is chosen because it isolates the averaging problem most cleanly. It is not representative of the full range of challenges governance systems face. In particular, the model does not address scenarios where centralized coordination provides genuine advantages — such as when a disturbance is truly system-wide and requires coordinated response across all nodes simultaneously.

The model does not capture learning or adaptation

Architecture A's controller uses fixed parameters throughout the simulation. Real institutions adapt: they update their models, reform their procedures, and improve their information systems over time. An important question the current model cannot address is whether high-latency, low-fidelity architectures can compensate for their structural disadvantages through institutional learning — and at what rate.

The adaptive controller extension (where gain adjusts dynamically based on observed performance) is a natural next development and would allow the simulator to address questions about institutional learning trajectories.

The comparison is between two idealized architectures

Architecture A and Architecture B represent extreme points in a continuous design space. Real governance systems are hybrids: partially centralized, partially distributed, with varying latency and fidelity at different layers. The simulation demonstrates the structural logic at the extremes; it does not map the intermediate

space where most real institutional design decisions are made.

This is a deliberate choice for clarity, not a claim that real systems are binary. The practical question is always about the direction of movement — whether a given reform increases or decreases effective latency, improves or degrades signal fidelity — rather than about achieving an idealized architecture.

What the simulator is not

The simulator is not a predictive model of any specific governance system. It does not take real-world data as input and produce forecasts. It does not prove that any particular institutional arrangement is superior in any particular context. It does not generate policy recommendations.

It is an analytical tool for understanding structural relationships. Its value is in making abstract principles — latency constraints, signal fidelity, the averaging problem — concrete and visualizable. The conclusions it supports are conclusions about structure, not about policy.

Part VI: Implications

The structural findings from the simulation generalize beyond the specific scenario modelled. This section draws out implications for governance design, for how governance failures are diagnosed, and for the broader project of treating institutional architecture as an engineering discipline.

Governance failures are often architectural misdiagnoses

When a governance system produces poor outcomes — slow crisis response, policies that harm the populations they are meant to serve, persistent drift away from stated goals — the standard diagnostic frameworks look for failures of competence, resources, political will, or corruption. These are real causes of governance failure and deserve serious attention.

But they are not the only causes, and they may not be the primary ones. A system that is architecturally incapable of perceiving its environment accurately, or of responding to it within the time window that crises allow, will produce poor outcomes regardless of the quality of the people operating it. Diagnosing such a system as a leadership failure, and responding by replacing leadership, is category error. The architecture will produce the same outputs with different people inside it.

The engineering framing makes this distinction tractable. It becomes possible to ask, for any observed governance failure: is this a parameter failure (the right architecture, poorly operated) or a structural failure (an architecture that cannot produce better outcomes given its constraints)? The answer shapes what interventions are appropriate.

The coordination layer is not optional

The finding that distributed systems require coordination protocols to avoid their own instability has a direct implication: pure decentralization is not the prescription that follows from the analysis. What follows is a specific architectural pattern — local decision authority operating within shared protocols established at a higher layer.

This distinction matters because the two failure modes it is trying to avoid pull in opposite directions. Excessive centralization produces the averaging problem: slow, uniform responses calibrated to distorted signals. Excessive decentralization without coordination produces fragmentation: local controllers that over-respond, interfere with each other, or optimize locally in ways that degrade the global system.

The stable architecture sits between these failure modes. Local nodes maintain high-fidelity observation of local conditions and respond with low latency. A coordination layer maintains shared protocols — what counts as a valid intervention, what the hard constraints are, what information must be shared laterally — without directing the content of local decisions. This is protocol-level governance rather than directive governance, and it has different structural properties from both pure centralization and pure decentralization.

Scale changes the problem

The averaging problem worsens as the number of nodes increases. A central controller managing ten nodes loses less spatial information than one managing a thousand. This means that governance architectures that are adequate at small scale may become structurally inadequate as the systems they govern grow more complex, more interconnected, and more differentiated.

This has a practical implication: governance architectures should be evaluated not just at their current scale but at the scale they will need to operate at as complexity increases. An architecture that is marginally stable under current conditions may become deeply unstable under foreseeable future conditions. The engineering approach permits this kind of prospective stability analysis, which the political framing does not.

Measurement is part of governance design

Signal fidelity is not a fixed property of a governance system — it is a design choice, or more precisely, a consequence of design choices made about what to measure, how to aggregate it, and how to transmit it to decision-makers.

This means that the information architecture of a governance system is as important as its decision architecture. A governance reform that improves institutional decision-making capacity without addressing the quality of the information flowing into those decisions will produce smaller improvements than one that addresses both. In some cases, improving information architecture alone — making previously invisible conditions legible, reducing aggregation loss, shortening the path from local observation to decision — may produce larger stability gains than any reform to the decision layer.

Economic and accounting systems are a special case of information architecture. What a society accounts for determines what its governance systems can see and respond to. Conditions that are not measured are, in the formal sense, unobservable — and unobservable conditions cannot be governed. The design of measurement and accounting systems is therefore governance design, whether or not it is recognized as such.

The phase transition in governance legitimacy

Historically, governance legitimacy has derived from two sources: authority (the right to govern, derived from tradition, divine mandate, or democratic consent) and ideology (the claim to know the correct direction, derived from political theory or moral philosophy).

The engineering framing suggests a third source: demonstrated performance. A governance architecture that can show, through transparent and reproducible methods, that it maintains stability more effectively and at lower cost than alternatives is making a legitimacy claim of a different kind — one that does not require agreement on values, only agreement on measurement.

This is not a claim that performance legitimacy should replace authority or ideological legitimacy. Governance involves irreducibly normative questions — about what to optimize for, whose stability matters, what counts as a crisis — that cannot be resolved by engineering analysis. But it is a claim that as the tools for demonstrating governance performance become more sophisticated and more accessible, the conversation about institutional design will increasingly be forced onto empirical terrain. Architectures that cannot demonstrate their performance will face growing pressure from those that can.

Governance as an engineering discipline

The deepest implication of this framework is disciplinary. Engineering disciplines are distinguished not by their subject matter but by their methods: they build formal models of the systems they study, test those models against observed behavior, and use the results to inform design decisions. The models are known to be simplifications. The simplifications are deliberate and documented. The goal is not perfect fidelity but actionable insight.

Governance has historically lacked this disciplinary infrastructure. Political science describes how governance systems work. Philosophy evaluates how they should work. History records how they have worked. But there is no mature discipline of governance engineering — the systematic application of formal modeling and empirical testing to institutional design questions.

The simulator presented here is a small step in this direction. The seven-primitive grammar, the state-space formulation, the reproducible simulation methodology — these are proposals for what a governance engineering toolkit might look like at its most basic level. Whether they prove useful will be determined by whether they generate insights that inform real design decisions, and by whether others find them extensible to questions the current framework cannot address.

Part VII: Conclusion

The argument made in this paper is deliberately narrow.

It does not claim that governance is reducible to engineering, or that the richness of political life — its normative complexity, its dependence on consent and legitimacy, its irreducibly human dimensions — can be captured in a state-space model. These claims would be false, and making them would undermine the specific claim that is true.

The specific claim is this: governance systems are feedback systems, and feedback systems have structural properties that determine their stability under disturbance. These properties can be modelled formally, compared objectively, and improved through design. Ignoring them does not make them go away. It just means that their consequences — slow crisis response, collateral disruption, persistent drift from target states — are attributed to the wrong causes and addressed with the wrong interventions.

The seven primitives introduced here — nodes, state, flows, latency, constraints, feedback, and signal fidelity — provide a minimal vocabulary for describing these structural properties. The simulator applies this vocabulary to a specific comparison and produces a specific finding: that localized crises are handled structurally better by architectures with local decision authority and high-fidelity local observation than by architectures that aggregate information centrally and respond uniformly. This finding has a name in control theory — it follows from Ashby's Law of Requisite Variety — and has been understood formally since the mid-twentieth century.

What is new here is not the mathematics. It is the application: taking tools developed for physical and engineering systems and applying them systematically to the design of governance institutions. And more specifically, building an open, reproducible, extensible simulator that makes these structural arguments not just assertable but demonstrable.

The work ahead is substantial. Empirical grounding — mapping real governance parameters onto the model's variables — would transform illustrative findings into testable hypotheses. Nonlinear extensions would capture the tipping-point dynamics that matter most in genuine crises. Heterogeneous network models would reflect the actual diversity of governance units. Adaptive controller models would address institutional learning. Each extension would also introduce new limitations, which would need to be documented with the same care as the current ones.

None of this work should be mistaken for value-neutral technocracy. Every design choice in a governance system embeds value judgments: about whose stability matters, what counts as a crisis, what constraints are non-negotiable. The engineering framing does not dissolve these questions. It sharpens them — by separating

the structural questions (given these goals and these constraints, what architecture can achieve them?) from the normative ones (what should the goals and constraints be?).

The people who first drew these diagrams — who mapped feedback loops and variety requirements and signal degradation — understood that they were doing something more than engineering. They were trying to find a language in which the structural necessities of viable complex systems could be made visible: not to win arguments, but to make certain kinds of mistakes harder to make.

That project is unfinished. This simulator is a small contribution to it.

Appendix A: Mathematical formulations

State transition equation

The core dynamics of each node follow a first-order discrete-time linear system with dead-time:

$$x(t+1) = A \cdot x(t) + B \cdot u(t - \tau) + d(t) + \text{drift}$$

Where:

- $x(t)$ — true state of the node at time t (scalar in single-node model, vector $\vec{x}(t)$ in multi-node model)
- A — natural decay coefficient (set to 0.95, representing slow entropy without intervention)
- B — actuator effectiveness (set to 1.0)
- $u(t - \tau)$ — control action applied τ steps ago (dead-time integration)
- $d(t)$ — external disturbance at time t
- $\text{drift} = x_{\text{ref}} \cdot (1 - A)$ — constant term that maintains equilibrium at x_{ref} in the absence of disturbance

Observation equation

The controller does not observe the true state directly. It observes a noisy measurement:

$$y(t) = x(t) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$$

Where σ is the standard deviation of observation noise. This models aggregation loss, reporting distortion, and measurement error. The gap between $y(t)$ and $x(t)$ is the observability deficit.

Control law

Both architectures use proportional feedback control:

$$u(t) = K \cdot (x_{\text{ref}} - y(t))$$

Where K is the controller gain and x_{ref} is the target equilibrium state.

Architecture A computes a single scalar control signal from the national mean of all node observations:

$$u_A(t) = K_A \cdot (x_{ref} - \text{mean}(\vec{y}(t)))$$

This uniform signal is then broadcast to all nodes, regardless of their individual conditions.

Architecture B computes a per-node control signal from each node's local observation:

$$u_{B,i}(t) = K_B \cdot (x_{ref} - y_i(t))$$

Stability ceiling

For a dead-time dominant discrete-time system, the approximate stability ceiling on controller gain is:

$$K_{max} \approx 1 / (\tau \cdot |A|)$$

Exceeding this ceiling produces oscillatory instability. The ceiling is lower for higher latency, which is why Architecture A uses a lower gain than Architecture B — not as a modelling choice, but as a stability requirement.

Multi-node coupling

In the ten-node model, adjacent nodes are coupled by a diffusion term representing crisis contagion:

$$\text{coupling}_i(t) = \beta \cdot \sum_{j \in \text{neighbours}(i)} (x_j(t) - x_i(t))$$

Where $\beta = 0.03$ is the coupling coefficient. The full state transition for the multi-node model is:

$$\vec{x}(t+1) = A \cdot \vec{x}(t) + \text{coupling}(\vec{x}(t)) + B \cdot \vec{u}(t - \tau) + \vec{d}(t) + \text{drift}$$

Performance metrics

Recovery time for node i is the number of time steps after the crisis until the node returns within a threshold of equilibrium:

$$RT_i = \min\{t > t_{crisis} : x_i(t) \geq x_{ref} - \delta\}$$

Where $\delta = 5$ in the current simulation.

Cumulative deficit for node i is the integral of stability loss below equilibrium after the crisis:

$$D_i = \sum_{t > t_{crisis}} \max(0, x_{ref} - x_i(t))$$

System-wide deficit is the sum across all nodes: $D_{total} = \sum_i D_i$.

Simulation parameters

Parameter	Architecture A	Architecture B
Latency τ	12	2
Observation noise σ	6.0	0.5
Controller gain K	0.30	0.45
Natural decay A	0.95	0.95
Actuator effectiveness B	1.0	1.0
Coupling coefficient β	0.03	0.03
Crisis magnitude	-45.0	-45.0
Crisis nodes	2, 7	2, 7
Number of nodes N	10	10
Time steps T	120	120
Crisis onset t_{crisis}	20	20

Appendix B: Code and reproduction

Source code

The simulator is implemented in Python using NumPy for numerical computation and Matplotlib for visualization. No dependencies beyond the standard scientific Python stack are required.

The full source code is available at:

github.com/BjornKennethHolmstrom/ggf-governance-simulator (<https://github.com/BjornKennethHolmstrom/ggf-governance-simulator>)

The repository includes:

- `ggf-simulator-v2.py` — single-node scalar model (latency and signal fidelity demonstration)
- `ggf-simulator-v3.py` — ten-node vector model (subsidiarity and the averaging problem)
- `README.md` — setup instructions and parameter documentation
- `/outputs` — pre-generated figures from the canonical parameter set

Reproducing the results

With Python 3.8+ and NumPy/Matplotlib installed:

```
git clone https://github.com/BjornKennethHolmstrom/ggf-governance-simulator
cd ggf-governance-simulator
python ggf-simulator-v3.py
```

The simulation is seeded for reproducibility (`numpy.random.default_rng(seed=7)`). Running with the default parameters reproduces the figures in this paper exactly.

Modifying the parameters

The architectural parameters are defined at the top of each script and are intended to be varied. Changing `tau_A`, `sigma_A`, `K_A` and their Architecture B counterparts will produce different quantitative outputs while preserving the qualitative structural relationships — provided gain values remain below the stability ceiling for their respective latencies.

Setting κ_B above approximately 0.5 for $\tau_B = 2$ will produce the oscillatory instability discussed in Part V. This behavior is intentional and informative: it demonstrates that the stability ceiling is a real constraint on distributed architectures as well as centralized ones.

Contributing

Extensions, critiques, and applications to specific governance contexts are welcome. The repository is open source under MIT license.

Appendix C: References and sources

A note on methodology

The concepts in this paper were developed through extended conversations with multiple AI systems — Claude (Anthropic), Gemini (Google), ChatGPT (OpenAI), DeepSeek, and Grok (xAI) — rather than through direct reading of the primary literature. The references below are the sources those systems identified as foundational to the ideas discussed. They are provided for readers who wish to engage with the primary literature directly, and to acknowledge the intellectual lineage of the framework honestly.

This is an unusual methodological position and worth being transparent about. The AI systems synthesized, connected, and in some cases extended these ideas in ways that shaped the specific formulations used here. The core mathematics — control theory, cybernetics, the Law of Requisite Variety — belongs to an established scientific tradition. The application to governance architecture, and the specific simulator implementation, emerged from this human-AI collaborative process.

Foundational sources

Wiener, N. (1948). *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press.

The foundational text of cybernetics. Wiener's explicit extension of feedback control concepts to biological and social systems is the intellectual origin of the approach taken here.

Ashby, W. R. (1956). *An Introduction to Cybernetics*. Chapman and Hall.

Contains the formal statement of the Law of Requisite Variety, which underlies the core argument about why centralized controllers cannot govern highly complex local environments. Available freely online via the Principia Cybernetica archive.

Ashby, W. R. (1952). *Design for a Brain*. Chapman and Hall.

Develops the concept of ultra-stability and adaptive systems — relevant to the adaptive controller extensions discussed in the limitations section.

Beer, S. (1972). *Brain of the Firm*. Allen Lane.

Beer's application of the Viable System Model to organizational governance. The most direct precedent for applying control-theoretic thinking to institutional design, including Beer's Cybersyn project in Chile — an early attempt at real-time national governance feedback systems.

Beer, S. (1979). *The Heart of Enterprise*. Wiley.

Develops the Viable System Model in detail, including the recursive structure of viable systems that prefigures the fractal / hierarchical governance architecture examined here.

Shannon, C. E., & Weaver, W. (1949). *The Mathematical Theory of Communication*. University of Illinois Press.

The foundation of information theory. The concept of signal fidelity used in this paper draws directly on Shannon's formalization of noise, channel capacity, and information loss.

Meadows, D. H., Meadows, D. L., Randers, J., & Behrens, W. W. (1972). *The Limits to Growth*. Universe Books.

A landmark application of systems dynamics to large-scale societal modeling. The intellectual tradition of treating civilization-scale systems as amenable to formal modeling.

Meadows, D. H. (2008). *Thinking in Systems: A Primer*. Chelsea Green Publishing.

An accessible introduction to systems dynamics and feedback loop analysis. The most readable entry point into the ideas that underlie the governance engineering approach.

Forrester, J. W. (1969). *Urban Dynamics*. MIT Press.

Forrester's application of system dynamics to urban governance — an early and controversial attempt to model the feedback structure of cities as formal control problems.

Control theory

Åström, K. J., & Murray, R. M. (2008). *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press.

The most accessible rigorous treatment of modern control theory. Available freely online. The stability analysis tools used in this paper — gain margins, dead-time effects, the separation principle — are covered here.

Franklin, G. F., Powell, J. D., & Emami-Naeini, A. (2019). *Feedback Control of Dynamic Systems*. Pearson.

Standard control engineering textbook. Reference for the gain ceiling approximation and dead-time stability analysis.

Related governance and complexity research

Ostrom, E. (1990). *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press.

Ostrom's empirical work on polycentric governance — communities self-organizing to manage shared resources — provides real-world evidence for the structural arguments made from theory here. Her design principles for robust common-pool resource institutions map closely onto the seven primitives.

Holland, J. H. (1995). *Hidden Order: How Adaptation Builds Complexity*. Addison-Wesley.

On complex adaptive systems and emergent behavior — relevant to the section on what governance systems cannot optimize directly.

Helbing, D. (2013). Globally networked risks and how to respond. *Nature*, 497, 51–59.

A systems-science perspective on cascading failures in globally coupled networks — directly relevant to the coupling and contagion dynamics modelled in the simulator.