$\eta^2 + \sigma^2 \approx 1$

A Control-Theoretic Theory of Intelligence at Inference Time

Joel Peña Muñoz Jr.

Contents

Preface

This book develops a control-theoretic framework for understanding and regulating intelligence at inference time. Rather than treating generative behavior as an intrinsic property of models, we formalize it as a dynamically regulated process governed by observable state variables. The central object of study is the empirically maintained constraint

$$\eta^2 + \sigma^2 \approx 1,$$

not as a physical law, but as a controlled manifold sustained through feedback.

Chapter 1

The Failure of Static Inference Control

1.1 Inference as an Open-Loop System

Autoregressive language model inference is typically executed as an open-loop process. Prior to generation, a fixed set of sampling hyperparameters—most commonly temperature (T) and nucleus sampling threshold (Top-p)—is selected. These parameters remain constant throughout the entire generation trajectory, regardless of changes in the model's internal probability landscape.

Formally, let $P_\theta(x_t \mid x_{<t})$ denote the conditional token distribution produced by a trained model with parameters θ. Standard inference applies a static transformation

$$\tilde{P}(x_t) = f(P_\theta(x_t); T, \text{Top-}p),$$

where $f(\cdot)$ is a nonlinear function that rescales logits and truncates the support. Once chosen, T and Top-p are not updated in response to the realized output statistics of $\tilde{P}$.

From a control-theoretic perspective, this constitutes an open-loop system. The operator specifies control inputs without observing the resulting system state, and no corrective action is taken if the output deviates from the intended behavior. Any mismatch between desired behavior (e.g., precision, diversity, coherence) and realized behavior persists unchecked for the duration of inference.

This design implicitly assumes that the mapping from hyperparameters to output behavior is stable, predictable, and monotonic. Empirically, this assumption fails. Small changes in temperature can produce discontinuous shifts in output structure, while identical hyperparameter settings can yield radically different behaviors depending on prompt structure,

token position, or latent model priors.

Crucially, the system provides no direct mechanism to observe whether the generated output exhibits the intended statistical properties. Concepts such as "creativity," "determinism," or "confidence" are inferred post hoc by human observers rather than measured during generation. As a result, standard inference offers no guarantees that a desired operating regime is maintained.

In control terms, autoregressive inference lacks:

- Observable state variables tied directly to output behavior,
- A feedback signal measuring deviation from a target state,
- A controller capable of correcting that deviation in real time.

The absence of feedback does not merely reduce robustness; it fundamentally limits what can be specified and enforced at inference time. Any apparent stability arises accidentally from model priors rather than from an explicit regulatory mechanism.

This failure motivates a reframing of inference not as parameter tuning, but as a dynamical system requiring closed-loop regulation.

1.2 Why Temperature and Top-p Are Indirect Proxies

Temperature (T) and nucleus sampling (Top-p) are the dominant mechanisms used to influence the behavior of autoregressive models at inference time. Despite their widespread use, neither parameter directly controls any observable statistical property of the output. Instead, both act as indirect modifiers of the token probability distribution, with effects that are model-dependent and context-sensitive.

Temperature rescales logits prior to normalization. Given logits z_i produced by the model, temperature-modified probabilities are computed as

$$P_T(x_i) = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}.$$

Lower values of T sharpen the distribution, while higher values flatten it. However, temperature does not specify how much uncertainty is introduced or removed; it merely rescales relative differences between logits. Two distributions with identical temperature can exhibit radically different entropy, mode dominance, or tail behavior.

Nucleus sampling further complicates this relationship. The Top-p procedure truncates the distribution by selecting the smallest subset of tokens whose cumulative probability

exceeds a threshold p, and renormalizing over that subset. Formally, this operation depends on the entire ordering of probabilities at each timestep. Small changes in the underlying distribution can cause discrete changes in the active token set, introducing non-smooth behavior in the resulting output statistics.

From a systems perspective, both T and Top-p are actuator variables, not state variables. They influence the system indirectly, without guaranteeing any specific outcome. There exists no fixed mapping

$$(T, \text{Top-}p) \longrightarrow \text{Desired Statistical Profile}$$

that holds across prompts, timesteps, or models. Consequently, operators must rely on trial-and-error tuning rather than principled specification.

This indirection produces three structural problems. First, intent is underspecified. Goals such as "increase creativity" or "increase precision" do not correspond to unique values of T or Top-p. Second, control is unstable. The same hyperparameters can yield different behaviors depending on prompt structure or generation depth. Third, failure is opaque. When generation collapses or becomes incoherent, the system provides no diagnostic signal indicating why.

Importantly, these limitations are not implementation flaws; they follow directly from the choice of control variables. Temperature and nucleus sampling modify the distribution without measuring it. They act blindly, assuming that local transformations will produce global effects aligned with intent.

Any framework that aims to regulate inference behavior must therefore abandon hyperparameter space as the primary control surface. Instead, control must be defined in terms of observable output statistics that directly encode the properties one wishes to maintain.

1.3 Nonlinearity Between Intent and Output Statistics

User intent at inference time is typically expressed in qualitative terms such as "more exploratory," "more deterministic," or "more precise." These intents implicitly refer to statistical properties of the generated output, yet the mapping from intent to output statistics is neither linear nor stable.

Let I denote an abstract intent variable supplied by an operator. In standard inference pipelines, I is mapped to a fixed set of hyperparameters $(T, \text{Top-}p)$. The resulting output statistics—entropy, mode dominance, token diversity, repetition rate—are emergent properties of the interaction between the model, the prompt, and the sampling procedure.

Formally, this process can be written as

$$I \longrightarrow (T, \text{Top-}p) \longrightarrow \tilde{P}(x_t) \longrightarrow S(\tilde{P}),$$

where $S(\tilde{P})$ denotes the realized statistical profile of the distribution.

This mapping is highly nonlinear. Small perturbations in $(T, \text{Top-}p)$ can produce disproportionate changes in $S(\tilde{P})$, while large changes may have negligible effect. The nonlinearity is amplified by the autoregressive nature of generation: early deviations in token selection alter the future conditioning context, compounding divergence over time.

Crucially, the same intent-to-hyperparameter mapping can yield distinct output statistics across different prompts or even across different positions within the same prompt. This context dependence violates a fundamental requirement of control: similar inputs should produce similar state trajectories. When this condition fails, stable regulation is impossible.

The problem is not merely that the system is sensitive; it is that intent is specified in a space orthogonal to the system's true state variables. Intent lives in semantic or task-level space, while control acts on logits. Without an explicit measurement layer linking the two, the system cannot correct deviations when the realized behavior diverges from the intended one.

As a result, operators routinely encounter pathological regimes: excessive repetition, premature collapse to a single mode, or uncontrolled dispersion. These regimes are often described as stochastic "quirks," but they are in fact predictable consequences of attempting to control a nonlinear dynamical system without feedback.

The presence of strong nonlinearities implies that any viable inference control framework must explicitly measure output statistics and adapt control signals in response. Static parameterization is structurally incapable of enforcing a target operating regime across contexts and time.

1.4 Motivation for Feedback Regulation

The limitations identified in the preceding sections arise from a single structural omission: the absence of feedback. Inference-time generation is currently performed without measuring whether the system's behavior aligns with the intended operating regime. Without feedback, there is no mechanism to correct deviations, suppress instability, or enforce consistency over time.

In control theory, feedback is introduced when a system exhibits nonlinearity, uncertainty, or sensitivity to initial conditions. Autoregressive generation exhibits all three. The

conditional probability distribution evolves at each timestep, shaped by both the prompt history and the model's internal priors. Consequently, the system state cannot be inferred from control inputs alone.

Introducing feedback requires two components: observable state variables and a control policy that adjusts actuators based on deviations from a target state. Crucially, the state variables must reflect the properties one seeks to regulate. Hyperparameters such as temperature and nucleus sampling thresholds do not meet this criterion, as they are not observations but interventions.

The motivation for feedback regulation at inference time is therefore not aesthetic or philosophical. It is a necessity imposed by the structure of the system. Without feedback, any apparent stability is accidental and fragile. With feedback, stability becomes an explicitly maintained property.

A feedback-regulated inference system enables the specification of behavior in terms of measurable statistical quantities rather than indirect tuning. Instead of asking which hyperparameters might produce a desired effect, the operator specifies a target operating regime, and the controller adjusts actuators to maintain it. Deviations caused by prompt structure, stochastic fluctuations, or internal model dynamics are corrected in real time.

This reframing shifts the locus of control from parameter selection to state regulation. Inference becomes a closed-loop dynamical process, in which the output distribution is continuously monitored and adjusted to remain within a prescribed envelope. Only under such a framework can claims about maintained trade-offs or controlled behavior be meaningfully evaluated.

Chapter 2 formalizes this shift by identifying which output statistics are admissible as state variables and why most commonly used metrics fail to satisfy the necessary constraints.

Chapter 2

Operationalizing Intelligence at Inference Time

2.1 Why Intelligence Must Be Measured, Not Interpreted

At inference time, intelligence is often described using interpretive language. Outputs are labeled as "intelligent," "confused," "creative," or "precise" based on human judgment after generation has completed. While such descriptions may be useful for qualitative evaluation, they are unsuitable as a foundation for regulation. Interpretation occurs outside the system; control must occur within it.

A system can only be regulated with respect to quantities that are measurable during operation. In control-theoretic terms, intelligence cannot be treated as a latent semantic property inferred post hoc. It must be decomposed into observable variables that can be computed from the system's instantaneous state. If a property cannot be measured in real time, it cannot be stabilized, tracked, or corrected.

Autoregressive models expose only one directly accessible object at inference time: the token probability distribution conditioned on the current context. All downstream behavior emerges from this distribution. Any claim about intelligence during generation must therefore reduce to a claim about properties of this distribution.

This constraint immediately excludes a large class of commonly cited metrics. Task success, factual correctness, coherence across long horizons, and human preference scores all require either ground-truth labels or full-sequence evaluation. They are retrospective and non-local. As such, they cannot serve as state variables for real-time regulation.

Interpretive judgments also introduce ambiguity. Two observers may disagree on whether

an output is "creative" or "precise," yet the system itself has no access to either judgment. Without a shared, quantitative definition, intent cannot be mapped consistently to system behavior. Control requires a single-valued signal, not a distribution of opinions.

Measurement resolves this ambiguity by enforcing operational definitions. A measured quantity is defined entirely by how it is computed. It admits bounds, supports comparison across contexts, and can be tracked over time. Most importantly, it can be used to compute an error signal: the difference between the current state and a desired target.

For inference-time intelligence, the relevant measurements must satisfy three conditions. First, they must be computable from the token distribution alone, without reference to future tokens. Second, they must be bounded and normalized to allow stable regulation. Third, they must correspond directly to properties that operators implicitly care about when they speak of exploration, determinism, or control.

The remainder of this chapter identifies which measurements satisfy these constraints and explains why the vast majority of intuitive or task-level metrics do not. This narrowing is not a limitation of the framework; it is the price of making inference-time regulation possible at all.

2.2 Constraints on Valid State Variables

Not every measurable quantity is suitable as a state variable for inference-time control. To function within a closed-loop regulatory system, a state variable must satisfy strict structural constraints. These constraints arise not from modeling preference, but from the requirements of stability, observability, and computability in real-time systems.

First, a valid state variable must be locally computable. Its value must be derivable from the model's instantaneous output distribution at a given timestep, without dependence on future tokens or full-sequence evaluation. Any metric requiring delayed aggregation or post hoc analysis cannot support feedback control, as it introduces latency that destabilizes the loop.

Second, the variable must be bounded. Unbounded quantities complicate error computation and can produce runaway control signals. Normalization to a fixed interval is therefore essential. Boundedness ensures that deviations are comparable across contexts and that controller gains can be selected without dependence on scale.

Third, the variable must vary smoothly under small perturbations of the underlying distribution. Discontinuous or highly quantized metrics introduce noise into the feedback signal, which can lead to oscillation or instability. While perfect smoothness is not required, the metric must respond predictably to incremental changes in the probability mass.

Fourth, the variable must be interpretable as a system property rather than a task outcome. Metrics such as accuracy or reward collapse multiple factors into a single scalar tied to external objectives. In contrast, a valid state variable describes how the system is behaving, not whether it has succeeded. Control acts on behavior; evaluation acts on outcomes.

Finally, the variable must be model-agnostic. Its definition should not depend on architectural details, training data, or task-specific annotations. A control framework tied to a single model class lacks generality and cannot support cross-system comparison or reuse.

Applying these constraints eliminates a large class of candidate metrics. Perplexity, log-likelihood over reference answers, human preference scores, and downstream task rewards all fail one or more criteria. They are either retrospective, unbounded, discontinuous, or externally defined.

What remains are metrics derived directly from the token probability distribution itself. These metrics describe the shape of the distribution—how concentrated it is, how dispersed it is, and how probability mass is allocated across alternatives. Such quantities are intrinsic to the inference process and available at every timestep.

The following sections formalize two such variables and justify their selection as the minimal sufficient state for regulating inference-time behavior.

2.3 Observability and Computational Feasibility

For a state variable to be usable in a feedback control loop, it must not only be well-defined and bounded, but also observable in practice. Observability here has a precise meaning: the state must be computable from signals already available to the system at inference time, within a latency budget that does not destabilize generation.

Autoregressive inference exposes a single high-dimensional signal at each timestep: the logit vector produced by the model prior to sampling. Any admissible state variable must be computable as a deterministic function of this vector, or of the corresponding probability distribution after normalization. No additional probes into internal activations or hidden states can be assumed, as such access is model-specific and often unavailable in deployed systems.

Computational feasibility imposes further constraints. Inference-time control operates on a per-token basis. State estimation must therefore be fast relative to token generation. Metrics that require quadratic or higher complexity in vocabulary size, or that rely on external models or evaluators, introduce unacceptable latency. Excessive delay converts feedback into lag, which degrades control performance and can induce oscillatory behavior.

These constraints rule out a variety of seemingly attractive approaches. Semantic sim-

ilarity metrics, classifier-based confidence scores, and reward-model evaluations all require additional forward passes or auxiliary networks. While useful for offline evaluation, they are incompatible with real-time regulation.

In contrast, metrics derived directly from the probability distribution are immediately observable. The normalization constant, cumulative probability mass, and extrema of the distribution are already computed as part of standard sampling procedures. Leveraging these quantities incurs minimal overhead and preserves compatibility with existing inference pipelines.

Observability also requires consistency across timesteps. A state variable whose definition changes implicitly with vocabulary size, tokenization scheme, or prompt length undermines comparability over time. Stable regulation demands that the same numerical value correspond to the same qualitative behavior regardless of context.

Taken together, these requirements sharply constrain the design space. The state variables used for inference-time control must be distributional, local, bounded, and inexpensive to compute. These conditions are not arbitrary design choices; they are dictated by the physics of real-time autoregressive generation.

The next section identifies which distributional properties satisfy these constraints and argues why only a minimal pair is required to characterize the relevant degrees of freedom.

2.4 Why Output Statistics Are the Only Accessible State

At inference time, the internal dynamics of a trained autoregressive model are largely opaque. While the model's parameters and hidden activations determine the shape of the output distribution, they are not directly accessible or interpretable as control variables during generation. What the system exposes at each timestep is the probability distribution over the next token, conditioned on the current context.

This distribution constitutes the full observable state of the system. Any attempt to regulate inference behavior must therefore operate on properties of this distribution alone. Internal representations, latent features, or semantic abstractions may explain why a distribution takes a particular form, but they cannot be acted upon in real time without model-specific instrumentation.

Output statistics provide a model-agnostic interface. Regardless of architecture, training procedure, or parameter count, an autoregressive model produces a normalized probability vector over a finite vocabulary. This common structure allows control mechanisms to be

defined independently of model internals, enabling portability across systems.

Crucially, output statistics capture the degrees of freedom that matter for inference-time behavior. Concentration of probability mass reflects determinism and collapse; dispersion reflects exploration and uncertainty. These properties directly influence downstream token selection and, by extension, the qualitative structure of generated text.

Attempts to regulate higher-level properties without referencing output statistics amount to controlling the system indirectly through unobserved variables. Such approaches inherit the same fragility as static hyperparameter tuning, as they lack a direct measurement of whether the intended behavior is being realized.

By restricting attention to output statistics, the control problem becomes well-posed. State estimation is immediate, bounded, and repeatable. Error signals can be computed at each timestep, and control actions can be applied without delay.

The remainder of this work focuses on identifying a minimal set of output statistics that fully characterize the controllable dimensions of inference-time behavior. Chapters 3 and 4 introduce these variables formally and justify their selection as sufficient state for feedback regulation.

Chapter 3

Output Dispersion (η)

3.1 Normalized Shannon Entropy

The first state variable introduced for inference-time regulation is Output Dispersion, denoted by η. This variable is designed to quantify how broadly probability mass is distributed across the vocabulary at a given timestep. Dispersion captures the degree to which the system is exploring alternatives rather than concentrating on a narrow set of tokens.

Formally, let $P(x)$ denote the normalized probability distribution over the vocabulary V produced by the model at a given timestep. The Shannon entropy of this distribution is defined as

$$H(P) = -\sum_{x \in V} P(x) \log P(x).$$

Entropy measures uncertainty in an information-theoretic sense. Higher entropy corresponds to greater dispersion of probability mass, while lower entropy indicates concentration.

Raw entropy alone is insufficient for control purposes because its magnitude depends on the size of the vocabulary. To remove this dependence and ensure boundedness, entropy is normalized by its maximum possible value, $\log |V|$. The resulting quantity is defined as

$$\eta = \frac{-\sum_{x \in V} P(x) \log P(x)}{\log |V|}.$$

By construction, $\eta \in [0, 1]$. A value of $\eta = 0$ corresponds to a degenerate distribution in which all probability mass is assigned to a single token. A value of $\eta = 1$ corresponds to a uniform distribution over the vocabulary. Intermediate values represent partial dispersion.

Normalization serves two critical functions. First, it allows η to be compared across models with different vocabulary sizes. Second, it enables stable control by fixing the dynamic range of the state variable. Without normalization, controller gains would need to be

retuned whenever the vocabulary or tokenization scheme changes.

From a control perspective, η is attractive because it varies smoothly with small changes in the underlying distribution. Incremental redistribution of probability mass produces incremental changes in entropy, avoiding the discontinuities that arise in metrics based on hard thresholds or truncation.

Importantly, η does not encode semantic diversity or task-level novelty. It measures only the shape of the distribution. This limitation is intentional. Dispersion, as defined here, is a structural property of inference, not an evaluative judgment about output quality.

As a state variable, η satisfies all admissibility criteria established in Chapter 2. It is locally computable, bounded, model-agnostic, and directly tied to the behavior of the sampling process. The next section examines why normalization is not merely convenient but essential for maintaining stability under feedback control.

3.2 Why Normalization Matters

Normalization is not a cosmetic adjustment to Shannon entropy; it is a structural requirement for stable feedback regulation. Without normalization, entropy scales with the logarithm of the vocabulary size, introducing variability that is unrelated to the behavior being controlled.

Consider two models with identical probability distributions over their respective vocabularies, but different vocabulary sizes. The unnormalized entropy values produced by these models will differ by a constant offset proportional to $\log |V|$. A controller that acts on raw entropy would therefore interpret identical dispersion as different system states, solely due to representational differences.

This dependence undermines any attempt to generalize control policies across models or tokenization schemes. Controller gains tuned for one vocabulary size would be miscalibrated for another, leading to either undercorrection or instability. Normalization eliminates this dependency by rescaling entropy into a fixed interval.

Normalization also simplifies error computation. In a feedback loop, control actions are driven by the difference between the current state and a target state. When the state variable lies in a known, bounded range, target specification becomes unambiguous. A desired dispersion of $\eta = 0.6$ has the same meaning regardless of context or model.

From a dynamical systems perspective, normalization ensures that the state space is compact. Compactness is a prerequisite for many stability guarantees. Without it, trajectories can drift arbitrarily far from nominal operating points, making regulation sensitive to noise and numerical error.

Normalization further enables interpretability of the control surface. The endpoints of

the interval correspond to clear limiting behaviors: complete concentration and maximal dispersion. Intermediate values interpolate smoothly between these extremes, allowing the operator to reason about trade-offs in a consistent coordinate system.

Finally, normalization is essential for coupling η with other state variables. When multiple states are combined in a control law or diagnostic quantity, differences in scale can distort their relative influence. By fixing $\eta \in [0, 1]$, its interaction with other normalized variables becomes well-defined.

For these reasons, normalized entropy is not merely a convenient choice; it is a necessary condition for treating dispersion as a controllable state variable. The following section examines the bounds of η in practice and clarifies how these theoretical limits manifest during inference.

3.3 Bounds and Interpretation

By definition, the normalized entropy η lies within the closed interval $[0, 1]$. These bounds are not merely theoretical extrema; they correspond to concrete and interpretable operating regimes of the inference process.

The lower bound, $\eta = 0$, occurs when the probability distribution collapses entirely onto a single token. In this regime, the system exhibits maximal concentration. All uncertainty is resolved prior to sampling, and the generation process becomes effectively deterministic. While such behavior may be desirable for short, unambiguous completions, sustained operation near this bound risks pathological repetition and loss of adaptability.

The upper bound, $\eta = 1$, corresponds to a uniform distribution over the vocabulary. In this regime, probability mass is maximally dispersed, and no token is preferred over any other. While this represents maximal exploration in a formal sense, it is rarely attainable in practice and generally undesirable. Uniformity erases the influence of learned structure, rendering generation incoherent and decoupled from context.

Most inference-time behavior occurs within the interior of the interval. Intermediate values of η represent partial dispersion, where a subset of tokens carries most of the probability mass, but alternatives retain non-negligible weight. This regime supports exploration while preserving sensitivity to model priors and context.

Importantly, the bounds of η are invariant under changes in prompt length, token position, or generation depth. A value of $\eta = 0.4$ represents the same degree of dispersion at the beginning of a response as it does several hundred tokens later. This time-consistency is essential for feedback control, as it allows targets to be specified globally rather than retuned dynamically.

The interpretation of η must remain structural. High dispersion does not imply novelty in a semantic or task-level sense, nor does low dispersion imply correctness or truth. The metric describes only how probability mass is allocated, not whether the resulting choice is appropriate. This distinction prevents misuse of η as a proxy for quality.

In practice, observed values of η cluster away from the extremes. Model priors, prompt constraints, and training distributions all bias the system toward moderate dispersion. These biases are not failures of the metric; they are reflections of the underlying system dynamics. A control framework must accommodate these tendencies rather than assume access to the full theoretical range.

The next section examines failure modes associated with entropy-based control and clarifies the limits of η as a standalone state variable.

3.4 Failure Modes of Entropy as a Proxy

While normalized entropy η satisfies the formal requirements of a controllable state variable, it does not fully characterize inference-time behavior on its own. Treating entropy as a complete description of system dynamics introduces failure modes that must be explicitly acknowledged.

The most immediate limitation is insensitivity to distribution shape beyond dispersion. Two probability distributions can exhibit identical entropy while differing substantially in structure. For example, a distribution with two dominant modes and a long tail may have the same entropy as a distribution with many moderately weighted alternatives. From the perspective of entropy alone, these states are indistinguishable, yet they produce different sampling behavior.

Entropy also fails to capture collapse within the high-probability region. A distribution in which one token carries a majority of the probability mass and the remainder is evenly distributed across a large tail can exhibit moderate entropy despite being effectively deterministic. In such cases, entropy overestimates exploration relative to the system's actual behavior.

Conversely, entropy can remain elevated even when the system exhibits unstable oscillatory behavior. Rapid shifts in which token holds the maximum probability can maintain dispersion while undermining coherence across timesteps. Entropy does not encode temporal consistency or persistence.

These failure modes arise because entropy measures aggregate uncertainty rather than dominance. It answers the question of how spread out the distribution is, but not how firmly the system is committed to any particular option. As a result, entropy alone cannot

distinguish between controlled exploration and uncontrolled fluctuation.

From a control perspective, reliance on a single scalar state is insufficient to stabilize a system with multiple degrees of freedom. The inference-time distribution possesses at least two independent axes of variation: dispersion and concentration. Entropy captures the former but is blind to the latter.

This observation motivates the introduction of a complementary state variable that measures stability directly. Chapter 4 introduces this variable and demonstrates how its interaction with entropy resolves the ambiguities identified here.

Chapter 4

Response Stability (σ)

4.1 Mode Mass as Determinism

To complement dispersion, we introduce a second state variable that captures concentration directly. This variable is intended to measure the system's commitment to a single outcome at a given timestep. We refer to this quantity as *Response Stability*, denoted by σ.

Let $P(x)$ denote the normalized probability distribution over the vocabulary V at a given timestep. Response Stability is defined as

$$\sigma = \max_{x \in V} P(x).$$

This quantity measures the probability mass assigned to the most likely token—the mode of the distribution.

By construction, $\sigma \in [1/|V|, 1]$. The lower bound corresponds to a uniform distribution, in which no token is preferred. The upper bound corresponds to total collapse, in which a single token is selected with probability one. Intermediate values represent partial dominance, where one option is favored but alternatives retain non-negligible mass.

Unlike entropy, which aggregates uncertainty across all tokens, σ isolates the dominant decision. It answers a different question: not how many alternatives exist, but how strongly the system is committed to one of them. This distinction is critical for inference- time control, as sampling behavior is disproportionately influenced by the leading mode.

From a behavioral standpoint, σ aligns closely with determinism. When σ is high, repeated sampling under identical conditions yields consistent outcomes. When σ is low, the system exhibits variability even if entropy remains moderate. This makes σ a direct proxy for stability under stochastic sampling.

Response Stability satisfies the admissibility criteria established in Chapter 2. It is locally

computable, bounded, model-agnostic, and inexpensive to evaluate. Computing σ requires only identification of the maximum probability value, an operation already performed implicitly in many sampling routines.

Importantly, σ does not depend on vocabulary size in the same way entropy does. While its theoretical lower bound varies with $|V|$, typical inference distributions are far from uniform, rendering this dependence negligible in practice. As a result, σ provides a robust signal across contexts.

The introduction of σ resolves the primary ambiguities identified in Chapter 3. Distributions that share identical entropy but differ in dominance are now distinguishable. The next section examines why alternative measures of stability fail to meet the same requirements.

4.2 Why Variance and Confidence Fail

Several alternative metrics are commonly proposed to capture stability or determinism in generative systems. Among the most frequently cited are variance-based measures, confidence scores, and logit margins. While each captures some aspect of distributional behavior, none satisfy the constraints required for inference-time control.

Variance-based measures quantify dispersion around an expected value. In the context of token distributions, however, variance is ill-defined. Tokens are categorical rather than ordinal, and any imposed ordering is arbitrary. As a result, variance depends on the chosen indexing scheme rather than intrinsic properties of the distribution. This violates the requirement of model-agnosticism and undermines interpretability.

Confidence scores often rely on auxiliary classifiers or heuristics that attempt to estimate the model's certainty about a prediction. These approaches introduce additional models into the loop, increasing computational cost and latency. More importantly, they collapse distributional information into a task-specific scalar, entangling behavior with external objectives. Such scores are evaluative rather than descriptive and cannot serve as neutral state variables.

Logit margins—the difference between the highest and second-highest logits—appear at first glance to be a natural proxy for determinism. However, margins are sensitive to scaling and normalization. A fixed margin can correspond to vastly different probability assignments depending on temperature and the distribution of lower-ranked logits. Margins also lack a natural bound, complicating normalization and controller design.

In contrast, mode mass directly measures what matters for sampling: the probability that the most likely token will be selected. This quantity is invariant under monotonic transformations of the logits and remains meaningful after normalization. It provides a

direct behavioral signal without reference to external labels or auxiliary models.

Another class of metrics attempts to infer stability from output sequences rather than from the instantaneous distribution. Repetition rates, self-BLEU scores, and n-gram diversity metrics fall into this category. While informative offline, they are retrospective and aggregate over multiple timesteps. They cannot support real-time regulation and introduce unavoidable delay into the control loop.

The failure of these alternatives reinforces a central theme of this work: inference-time control demands simplicity and immediacy. Metrics must describe the system's current state, not its past performance or future utility. Mode mass meets this requirement with minimal assumptions.

The following section examines the bounds of σ in practice and clarifies how it behaves under realistic inference conditions.

4.3 Lower and Upper Bounds

The Response Stability metric σ is bounded by construction, but its practical behavior under inference warrants closer examination. Understanding these bounds is essential for interpreting σ as a control variable rather than a purely descriptive statistic.

The theoretical lower bound of σ is $1/|V|$, corresponding to a perfectly uniform distribution over the vocabulary. In this regime, no token is preferred, and sampling reduces to random selection. While this bound is well-defined, it is rarely approached in practice. Trained autoregressive models encode strong structural priors that bias probability mass toward a limited subset of tokens at each timestep. As a result, observed values of σ are typically orders of magnitude larger than the uniform baseline.

The upper bound, $\sigma = 1$, corresponds to total mode collapse. In this regime, the system assigns all probability mass to a single token, rendering generation fully deterministic. Such behavior can arise intentionally, for example under extremely low temperature, but more often emerges as a pathological state. Sustained operation near this bound eliminates adaptability and can induce repetitive or degenerate output patterns.

Between these extremes lies the operational regime of interest. Intermediate values of σ indicate partial dominance, where one token is favored but alternatives retain non-zero probability. This regime supports controlled stochasticity: repeated sampling yields similar but not identical outputs, balancing consistency with variability.

Unlike entropy, the bounds of σ have a direct behavioral interpretation. Changes in σ translate immediately into changes in sampling outcomes. A small increase in σ increases the likelihood of repeated selections of the same token, while a decrease introduces variability

even when the overall distribution remains concentrated.

It is important to note that the effective range of σ is constrained by model priors and context. Certain prompts or syntactic positions naturally produce high stability, while others induce dispersion regardless of control inputs. These constraints are not violations of the metric; they reflect the underlying dynamics of the trained model.

For control purposes, the relevant question is not whether the theoretical bounds are reachable, but whether σ varies smoothly and predictably within the accessible range. Empirically, mode mass responds monotonically to changes in sampling actuators, making it suitable for feedback regulation.

The next section examines how σ interacts with model priors and why stability cannot be fully imposed externally without accounting for internal biases.

4.4 Interaction Between σ and Model Priors

Response Stability σ does not operate in isolation. Its behavior is shaped by the model's internal priors, which encode statistical regularities learned during training. These priors exert a persistent influence on the probability distribution, constraining how stability responds to external control inputs.

Model priors manifest as structured biases in the output distribution. Syntactic rules, frequent token sequences, and learned patterns of continuation all increase the probability of certain tokens relative to others. In regions of the state space where priors are strong, σ may remain elevated even when sampling actuators are adjusted to encourage dispersion.

This interaction reveals a critical asymmetry. External control can reduce stability only to the extent permitted by the model's learned structure. Attempts to force σ below this implicit floor result in diminished responsiveness rather than smooth continuation of the trend. The controller may increase temperature or expand the sampling support without substantially lowering the dominant mode probability.

Conversely, increasing σ is generally easier than decreasing it. Many prompts and contexts naturally favor a small set of likely continuations. In such cases, modest changes to control inputs can rapidly push the system toward near-deterministic behavior. This asymmetry has implications for controller design, particularly in selecting gain parameters that avoid overshoot or premature collapse.

The dependence of σ on model priors underscores an important limitation of inference-time regulation: control cannot override learned structure entirely. The controller shapes how the system navigates its probability landscape, but it does not redefine that landscape. Strong priors act as competing forces, effectively functioning as internal controllers with their

own objectives.

Recognizing this interaction prevents misattribution of failure. When stability persists despite aggressive control, the cause is not a breakdown of the metric but a reflection of the model's internal dynamics. A well-designed control framework must therefore operate within an envelope defined jointly by external actuators and internal priors.

With both dispersion and stability now defined and contextualized, the stage is set to examine their joint behavior. Chapter 5 analyzes how these two state variables interact and why their coupled dynamics trace a constrained manifold under feedback regulation.

Chapter 5

The Emergence of the Trade-Off Curve

5.1 Empirical Observation of Coupled Dynamics

With dispersion η and stability σ defined as independent state variables, their joint behavior can now be examined empirically. When measured across a range of inference conditions, these variables do not vary independently. Instead, they exhibit a structured dependence that constrains the system's accessible state space.

Empirical measurements were obtained by sampling the output probability distribution at each timestep under controlled variation of inference parameters. For each configuration, both η and σ were computed directly from the distribution. When plotted against one another, the resulting points do not fill the η–σ plane uniformly. Rather, they concentrate along a curved trajectory.

This concentration indicates coupling between dispersion and stability. Increases in dispersion are systematically accompanied by decreases in stability, and vice versa. However, the relationship is not linear. Regions of moderate dispersion correspond to disproportionately large reductions in stability, while further increases in dispersion produce diminishing returns.

Importantly, this coupling persists across prompts, timesteps, and sampling runs, provided the system operates under normal inference conditions. While the exact location of points along the curve varies with context, the overall shape of the trajectory remains consistent. This suggests that the observed relationship is not an artifact of a particular prompt or task, but a property of the regulated inference process.

The absence of points far from the curve is as informative as the presence of points on

it. Large regions of the η–σ plane appear inaccessible. For example, states with both high dispersion and high stability are rarely observed, as are states with low dispersion and low stability. The system naturally avoids these combinations under standard operation.

At this stage, no claim is made that the curve represents a fundamental law. It is presented only as an empirical regularity. The significance of this observation lies in its repeatability and in its implication that dispersion and stability are not free parameters, but constrained degrees of freedom.

The next section examines the origin of this curvature and explains why linear changes in control inputs produce nonlinear trajectories in the state space.

5.2 Why Linear Control Produces Curved Manifolds

The empirical curvature observed in the η–σ plane is not accidental. It arises from the interaction between linear control inputs and nonlinear transformations within the inference pipeline. Understanding this mechanism is essential for interpreting the observed trade-off correctly.

Control inputs such as temperature and nucleus sampling thresholds are typically adjusted linearly with respect to a scalar intent parameter. That is, a change in intent produces a proportional change in actuator values. However, the mapping from these actuators to the resulting probability distribution is fundamentally nonlinear.

Temperature rescales logits exponentially prior to normalization. Small linear changes in temperature can therefore produce large nonlinear changes in relative probabilities, especially in regions where logits are already separated by significant margins. Similarly, nucleus sampling introduces discrete truncation effects that alter the support of the distribution in a non-smooth manner.

These nonlinearities propagate into the state variables. Entropy responds to changes in the entire distribution, integrating effects across all tokens. Mode mass, by contrast, responds primarily to changes near the maximum. As control inputs vary, the two metrics evolve at different rates, producing curved trajectories in their joint state space.

From a dynamical systems perspective, the inference process can be viewed as a nonlinear mapping

$$u \mapsto (\eta(u), \sigma(u)),$$

where u denotes the vector of control inputs. Even if u varies along a straight line, its image under this mapping need not be linear. Curvature in the image reflects the underlying geometry of the transformation.

The appearance of a smooth curve rather than a scattered cloud indicates that the system is operating within a constrained manifold. The controller does not independently set η and σ; it navigates a path determined by the interaction between actuators, model priors, and normalization effects.

This perspective clarifies why attempts to tune dispersion and stability independently often fail. Linear adjustments to control parameters do not correspond to linear movements in state space. Without accounting for this geometry, operators misinterpret the system's responsiveness and attribute curvature to noise rather than structure.

The following section formalizes this geometric interpretation and introduces a diagnostic quantity that captures the observed constraint succinctly.

5.3 Geometric Interpretation of η–σ Space

The coupled evolution of η and σ admits a natural geometric interpretation. Rather than occupying a rectangular state space in which each variable can be adjusted independently, inference-time behavior is constrained to a lower-dimensional manifold embedded within the η–σ plane.

Each point in this plane represents a possible instantaneous configuration of the output distribution. The empirical observations described in the previous section indicate that the system's trajectory under normal operation lies close to a smooth curve. This curve acts as an attractor for the controlled dynamics, concentrating the system's behavior along a narrow path.

Geometrically, this implies that η and σ are not orthogonal degrees of freedom. Changes in one induce compensatory changes in the other. The curvature of the trajectory encodes the trade-off between dispersion and stability imposed by the inference mechanism and the controller.

The absence of reachable states far from the curve suggests the presence of implicit constraints. These constraints are not enforced explicitly by the controller, but emerge from the combined effects of normalization, probabilistic sampling, and model priors. The controller navigates within this constrained geometry rather than reshaping it.

This geometric view also clarifies why scalar summaries of inference behavior are often misleading. Reducing the system's state to a single metric discards information about where the system lies along the trade-off curve. Two states with identical entropy but different stability correspond to distinct positions in η–σ space, with different sampling behavior.

Interpreting inference-time behavior geometrically shifts emphasis from parameter tuning to trajectory management. The goal of control is not to fix η or σ independently, but to

place the system at an appropriate location along the admissible manifold and keep it there despite perturbations.

This perspective sets the stage for introducing a scalar diagnostic that summarizes the system's position relative to the observed curve without collapsing the underlying structure. The next section introduces such a diagnostic and clarifies its role.

5.4 The Diagnostic Quantity $\eta^2 + \sigma^2$

The geometric structure observed in η–σ space motivates the introduction of a scalar diagnostic that summarizes the system's position relative to the empirical manifold. This diagnostic is not intended as a fundamental invariant or conservation law. It is a compact descriptor of a maintained operating regime under feedback regulation.

Empirically, points sampled during controlled inference cluster near a curve that is well approximated by the relation

$$\eta^2 + \sigma^2 \approx 1.$$

This relation describes a quarter-circle in the first quadrant of the η–σ plane. The approximation captures the dominant geometry of the observed trade-off without implying exact equality.

Interpreted diagnostically, the quantity

$$\Sigma = \eta^2 + \sigma^2$$

serves as a scalar measure of adherence to the controlled manifold. Values of Σ near unity indicate that the system remains within the expected operating envelope. Deviations from unity indicate that the coupling between dispersion and stability has weakened or broken.

It is critical to emphasize that Σ has no independent operational meaning. It does not define a target for control, nor does it prescribe desirable behavior. Its sole purpose is to provide a low-dimensional indicator of whether the system's joint state lies on or off the empirically observed trajectory.

The usefulness of Σ derives from its sensitivity to decoupling. When external noise, adversarial prompting, or internal model dynamics disrupt the usual relationship between η and σ, the diagnostic value shifts away from unity. Such shifts are difficult to detect by inspecting either variable in isolation.

From a control perspective, Σ functions as a consistency check rather than a control objective. The controller regulates η and σ indirectly via actuator adjustment. Σ provides a means of monitoring whether those adjustments preserve the expected geometry of the state

space.

Importantly, the approximate constancy of Σ is contingent on operating conditions. It holds under normal inference with feedback regulation and degrades under perturbation. This conditionality distinguishes the diagnostic from a physical law and preserves the framework's falsifiability.

Chapter 6 formalizes the control architecture that maintains trajectories along this manifold and explains how deviations are corrected in real time.

Chapter 6

Feedback Control at Inference Time

6.1 Closed-Loop vs Open-Loop Generation

The distinction between open-loop and closed-loop generation is foundational to the control architecture developed in this work. Open-loop generation, as described in Chapter 1, operates by fixing control inputs prior to inference and applying them uniformly throughout the generation process. In contrast, closed-loop generation continuously monitors the system state and adjusts control inputs in response to observed deviations.

Formally, an open-loop inference process can be represented as

$$u_t = u_0 \quad \forall t,$$

where u_t denotes the vector of control inputs at timestep t. The system evolves according to its internal dynamics, but no information about the realized output statistics feeds back into the control policy.

Closed-loop generation introduces state dependence. Let $x_t = (\eta_t, \sigma_t)$ denote the measured state at timestep t. Control inputs are then computed as a function of this state:

$$u_t = g(x_t),$$

where $g(\cdot)$ is a control policy designed to regulate the system toward a desired operating regime.

The introduction of feedback fundamentally alters the behavior of the inference process. Rather than drifting according to uncontrolled nonlinear dynamics, the system actively corrects deviations induced by stochastic sampling, prompt variation, or internal model priors. Stability is no longer an incidental byproduct of training but an explicitly maintained prop-

erty.

Closed-loop generation also enables time-consistent behavior. In open-loop systems, early timesteps and late timesteps may exhibit qualitatively different statistical profiles under identical hyperparameters. Feedback mitigates this drift by re-estimating the state at each timestep and applying corrective adjustments as needed.

Importantly, closed-loop control does not eliminate stochasticity. Instead, it constrains stochastic behavior within a prescribed envelope. The system remains probabilistic, but its statistical properties are regulated rather than left to chance. This distinction preserves flexibility while preventing collapse or uncontrolled dispersion.

The transition from open-loop to closed-loop inference reframes generation as a dynamical system under active regulation. The remaining sections of this chapter specify how user intent is mapped into control targets and how control actions are computed to maintain desired trajectories in η–σ space.

6.2 Control Parameterization via User Intent

A closed-loop inference system requires a mechanism for specifying desired behavior. This specification must be simple enough to be usable, yet precise enough to map onto measurable system states. In this framework, user intent is represented as a single scalar control parameter, denoted by $\lambda \in [0, 1]$.

The role of λ is not to directly encode semantic goals or task-level objectives. Instead, it parameterizes a target location along the admissible η–σ manifold. Low values of λ bias the system toward high stability and low dispersion, while high values bias it toward high dispersion and low stability. Intermediate values interpolate between these extremes.

Formally, λ defines a desired operating regime rather than a fixed target state. The controller does not attempt to enforce exact values of η or σ. Instead, it adjusts control inputs to keep the system near the region of the manifold associated with the chosen value of λ.

This parameterization resolves a central ambiguity in standard inference interfaces. Qualitative directives such as “be more precise” or “be more exploratory” are mapped to a single continuous variable with a well-defined interpretation. The operator specifies intent in a low-dimensional space, and the controller handles the complexity of translating that intent into actuator adjustments.

Crucially, λ is decoupled from specific hyperparameters. It does not correspond to a fixed temperature or nucleus sampling threshold. Instead, it indexes a family of control actions whose effect is mediated through feedback. This decoupling allows the same intent

parameter to produce consistent behavior across prompts, timesteps, and models.

From a control perspective, λ functions as a reference signal. The measured state (η_t, σ_t) is compared implicitly to the regime associated with λ, and control actions are computed to reduce deviation. The precise mapping between λ and actuator values is implementation-dependent and may be tuned empirically.

By reducing user input to a single scalar, the framework balances expressiveness with robustness. Higher-dimensional intent specifications risk overfitting and instability, while a scalar parameter provides a smooth, interpretable control surface.

The next section specifies the control logic used to translate deviations in the measured state into actuator adjustments and examines its stability properties.

6.3 PI Control Logic

With a reference parameter λ specified and state variables (η_t, σ_t) observable at each timestep, the remaining task is to define a control policy that translates state deviations into actuator adjustments. The controller must be simple, stable, and robust to noise. For these reasons, a proportional–integral (PI) control structure is adopted.

The objective of the controller is not to enforce exact values of η or σ, but to regulate the system so that its trajectory remains near the region of the η–σ manifold associated with the chosen λ. To achieve this, an error signal is defined in terms of dispersion alone, with stability responding implicitly through the coupled dynamics.

Let $\eta^*(\lambda)$ denote the target dispersion corresponding to the desired operating regime. At timestep t, the instantaneous error is

$$e_t = \eta^*(\lambda) - \eta_t.$$

This error measures deviation from the desired level of dispersion and serves as the primary feedback signal.

The proportional component of the controller responds to the current error,

$$u_t^{(P)} = K_P \, e_t,$$

where K_P is a proportional gain. This term provides immediate correction, increasing or decreasing actuator values in response to deviations.

The integral component accumulates error over time,

$$u_t^{(I)} = K_I \sum_{\tau=0}^{t} e_\tau,$$

where K_I is an integral gain. This term compensates for persistent biases induced by model priors or prompt structure, ensuring that steady-state deviations are eliminated.

The total control signal is given by

$$u_t = u_0 + u_t^{(P)} + u_t^{(I)},$$

where u_0 denotes a nominal actuator setting associated with λ. The controller modulates this baseline rather than replacing it, preserving continuity with standard sampling behavior.

The choice of a PI controller reflects a deliberate trade-off. Purely proportional control is insufficient to counteract persistent biases, while derivative control introduces sensitivity to noise in the measured state. PI control offers a balance between responsiveness and stability, well-suited to the stochastic environment of inference.

Stability of the closed-loop system depends on the selection of gains K_P and K_I. Gains that are too large can induce oscillation, while gains that are too small result in sluggish response. However, because η is bounded and normalized, gain selection is tractable and does not require model-specific scaling.

Although the controller operates directly on dispersion, stability σ responds indirectly through the coupled dynamics established in Chapter 5. This indirect regulation is sufficient to maintain trajectories along the observed manifold without requiring explicit control of both variables.

The next section specifies how the abstract control signal u_t is mapped onto concrete sampling actuators and examines the resulting stability properties.

6.4 Actuator Scheduling and Stability

The control signal produced by the PI controller must ultimately be translated into concrete sampling actions. This translation occurs through actuator scheduling, in which the abstract control variable u_t modulates inference hyperparameters such as temperature and nucleus sampling thresholds.

Let u_t be a scalar control signal derived from the PI logic described in the previous

section. Actuator scheduling defines deterministic mappings

$$T_t = f_T(u_t), \qquad p_t = f_p(u_t),$$

where T_t denotes temperature and p_t denotes the nucleus sampling threshold at timestep t. These mappings are chosen to be monotonic and bounded, ensuring that control actions remain within safe operational limits.

The use of bounded actuator ranges is essential for stability. Temperature values below a minimum threshold risk immediate mode collapse, while values above a maximum threshold destroy sensitivity to model priors. Similarly, extreme nucleus thresholds either truncate too aggressively or admit excessive noise. Scheduling functions therefore enforce hard constraints that prevent the controller from driving the system into degenerate regimes.

Stability of the closed-loop system depends on the combined dynamics of the controller, the actuator mappings, and the inference process itself. While a full analytical treatment is intractable due to the high dimensionality of the underlying model, qualitative stability can be assessed through boundedness and monotonicity. Because η is normalized and bounded, the error signal remains finite. Because actuator mappings are bounded, control inputs cannot diverge. Together, these properties prevent runaway behavior.

Empirically, actuator scheduling smooths the effect of control actions. Rather than producing abrupt changes in sampling behavior, gradual adjustments maintain continuity in the output distribution. This continuity reduces oscillation and supports convergence toward a steady operating regime.

It is important to note that stability here does not imply convergence to a fixed point. The system operates under stochastic sampling and time-varying context. Stability instead means confinement within a bounded region of state space corresponding to the desired operating envelope. Deviations occur, but they are corrected rather than amplified.

By coupling bounded actuator scheduling with PI control over normalized dispersion, the framework achieves practical regulation of inference-time behavior. The resulting dynamics maintain trajectories along the empirical η–σ manifold identified in Chapter 5, without requiring explicit enforcement of the diagnostic relation.

With the control architecture fully specified, the next chapter examines conditions under which this regulation fails and explains why such failures are essential for validating the framework.

Chapter 7

Falsification and Failure

7.1 Noise Injection

A control framework that claims to regulate inference-time behavior must remain falsifiable. One of the most direct methods for testing such a system is controlled noise injection. Noise serves as an external perturbation that disrupts the assumed relationship between control inputs and observed state variables, revealing whether observed stability is intrinsic or maintained.

In this context, noise injection is performed by adding stochastic perturbations to the logits prior to normalization. Let z_i denote the original logits produced by the model at a given timestep. Noise injection modifies these logits according to

$$\tilde{z}_i = z_i + \epsilon_i,$$

where $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$ is Gaussian noise with variance σ_n^2.

This perturbation alters the probability distribution in a manner that is independent of the controller's intent. Because noise is added prior to normalization, it can both flatten and reshuffle the distribution, affecting entropy and mode mass in ways that are not predictable from control inputs alone.

When noise injection is mild, the feedback controller compensates by adjusting actuators to restore the target operating regime. Dispersion and stability fluctuate but remain coupled, and the diagnostic quantity $\Sigma = \eta^2 + \sigma^2$ remains close to unity. This behavior demonstrates robustness of the control loop to small perturbations.

As noise magnitude increases, compensation becomes ineffective. The relationship between η and σ degrades, and measured states drift away from the empirical manifold. Notably, entropy may increase without a corresponding decrease in stability, or stability may

collapse without proportional dispersion. In these regimes, the diagnostic quantity drops significantly below its nominal value.

This decoupling is critical. It demonstrates that the observed relation between η and σ is not an inherent property of the model. If it were, perturbations to the logits would preserve the relationship. Instead, noise injection breaks the coupling, confirming that the maintained geometry is a product of the control architecture.

Noise injection thus serves a dual role. It validates the controller under realistic perturbations and falsifies any claim that the diagnostic relation represents a universal law. The framework predicts both behaviors and distinguishes clearly between them.

The next section examines a different class of perturbations—adversarial prompts—and shows how internal model dynamics can overpower external control.

7.2 Adversarial Prompting

Noise injection perturbs the inference process externally, without altering the semantic or structural content of the prompt. Adversarial prompting introduces a different class of stress test by exploiting internal model dynamics. Rather than injecting randomness into the logits, the prompt itself is constructed to induce pathological behavior.

Adversarial prompts are designed to activate strong model priors that dominate the output distribution. Common examples include repetitive loop-inducing instructions, syntactically degenerate continuations, or prompts that encourage self-referential recursion. In these cases, the model's learned structure exerts a force that competes directly with the external controller.

Under adversarial prompting, the measured dispersion η and stability σ can exhibit behavior that deviates sharply from the regulated manifold. In particular, stability may increase toward unity even when the controller attempts to maintain high dispersion. Entropy may decrease only marginally, producing states with high σ and moderate η that lie far from the empirical trajectory.

This behavior highlights a fundamental limitation of inference-time control. The controller operates on the output distribution, but it does not modify the underlying probability landscape defined by the model's parameters. When internal priors strongly favor a narrow set of continuations, external control inputs have limited leverage.

Importantly, adversarial prompting does not cause random failure. The resulting deviations are structured and repeatable. Given the same prompt and control settings, the system consistently collapses into similar states. This repeatability indicates that the failure is driven by deterministic internal dynamics rather than stochastic noise.

From the perspective of the diagnostic quantity $\Sigma = \eta^2 + \sigma^2$, adversarial prompting produces a characteristic signature. The diagnostic value departs from unity in a systematic manner, reflecting decoupling between dispersion and stability. Unlike noise injection, which introduces scatter, adversarial prompts drive the system toward specific off-manifold regions.

These observations reinforce a key distinction. The regulated manifold is not a universal constraint imposed by the model. It is a maintained operating regime that holds only when external control and internal dynamics are aligned. When they conflict, internal dynamics prevail.

The final section of this chapter synthesizes these failure modes and explains why they are essential to the credibility of the framework.

7.3 Model Priors as Competing Controllers

The failure modes observed under adversarial prompting can be understood by reframing model priors as implicit controllers embedded within the system. These priors, learned during training, encode preferences over token sequences that exert persistent influence during inference. They operate continuously, shaping the probability distribution independently of external control inputs.

From a control-theoretic perspective, this introduces a multi-controller system. The inference-time controller adjusts sampling actuators to regulate observable state variables, while the model's internal priors push the distribution toward patterns that maximize likelihood under the training objective. When these controllers are aligned, regulation is effective. When they conflict, the stronger controller dominates.

Model priors differ from external controllers in two critical ways. First, they act directly on the logits prior to any sampling transformation. Second, they are optimized over long timescales and large datasets, giving them substantial leverage in regions of the state space associated with common or reinforced patterns. External control operates downstream, with limited authority over the shape of the distribution.

This asymmetry explains why certain failures are resistant to correction. In repetitive or self-referential prompts, the model's priors strongly favor a narrow continuation manifold. The external controller may increase temperature or expand the sampling support, but these actions only perturb the distribution locally. The dominant mode remains entrenched, and stability persists.

Interpreting priors as competing controllers clarifies the scope of inference-time regulation. Control does not override learned behavior; it negotiates with it. The regulated manifold described in Chapter 5 emerges only when the external controller operates within

the envelope permitted by internal dynamics.

This interpretation also reframes apparent fragility as a diagnostic feature. When the system departs from the expected η–σ relationship, the deviation indicates not a failure of measurement or control logic, but the presence of strong internal forces. The diagnostic quantity highlights these conflicts rather than concealing them.

Recognizing model priors as controllers prevents overextension of the framework. It establishes clear boundaries on what inference-time control can and cannot achieve. Within those boundaries, regulation is effective and predictable. Outside them, failure is systematic and informative.

The next section synthesizes the observed failure modes and explains why their existence is essential to the framework's falsifiability.

7.4 Why Failure Confirms the Framework

The presence of systematic failure modes is not a weakness of the proposed framework; it is a necessary condition for its scientific validity. A control architecture that could not be forced to fail would be unfalsifiable and therefore uninformative. The observed breakdowns under noise injection and adversarial prompting serve as critical tests of scope and mechanism.

If the relation between dispersion η and stability σ were an intrinsic property of the model, perturbations to the logits or prompt structure would preserve that relation. Empirically, they do not. Instead, the diagnostic quantity $\Sigma = \eta^2 + \sigma^2$ deviates from its nominal value in predictable ways under specific stressors. This behavior confirms that the observed coupling is maintained by control rather than imposed by the model.

Failure also differentiates controlled behavior from coincidence. Under normal operation, the system exhibits a stable trajectory in η–σ space. Under perturbation, that trajectory deforms or dissolves. The ability to induce and characterize this transition demonstrates causal influence of the controller on the system's behavior.

Moreover, failure reveals the limits of regulation. Noise injection exposes sensitivity to external randomness, while adversarial prompting exposes the dominance of internal priors. Each failure mode maps to a distinct mechanism, allowing the framework to distinguish between external disturbances and internal constraints. This diagnostic resolution would be impossible if the framework relied on opaque or task-level metrics.

From a control perspective, the framework succeeds precisely because it does not promise universal stability. Instead, it specifies the conditions under which stability is achievable and predicts the form of breakdown when those conditions are violated. Such conditional claims are testable and falsifiable.

Failure therefore functions as evidence. It validates the interpretation of η and σ as regulated state variables, confirms the role of feedback in maintaining the observed manifold, and delineates the boundaries of inference-time control.

With the framework's limits established, the next chapter reframes inference as a dynamical system and situates the control architecture within a broader systems-theoretic context.

Chapter 8

Inference as a Dynamical System

8.1 State Space Interpretation

With the control architecture and its failure modes established, inference-time generation can now be reframed explicitly as a dynamical system evolving in a low-dimensional state space. This reframing provides a unifying perspective that connects measurement, control, and observed behavior.

At each timestep t, the system occupies a state

$$x_t = (\eta_t, \sigma_t),$$

where η_t and σ_t are computed directly from the output probability distribution. This state summarizes the aspects of the distribution relevant to inference-time behavior. Although the underlying model operates in a high-dimensional parameter space, the observable dynamics relevant to control are effectively projected onto this two-dimensional manifold.

The evolution of the state is governed by a combination of internal and external factors. Internal dynamics arise from the model's learned priors and the autoregressive conditioning on previously generated tokens. External dynamics arise from control actions that modulate sampling behavior. Together, these influences define a discrete-time dynamical system of the form

$$x_{t+1} = F(x_t, u_t, c_t),$$

where u_t denotes control inputs and c_t denotes contextual factors such as the prompt history.

In the absence of control, the system evolves according to its internal dynamics alone. Trajectories may drift, collapse, or oscillate depending on prompt structure and model biases. With feedback control, trajectories are confined to a bounded region of state space,

corresponding to the regulated operating envelope described in earlier chapters.

Viewing inference through the lens of state space clarifies the meaning of stability. Stability does not imply convergence to a fixed point. Instead, it refers to confinement within a region of state space despite stochastic perturbations and time-varying context. The regulated manifold identified in Chapter 5 functions as an invariant set under nominal conditions, attracting trajectories back toward it when deviations occur.

This interpretation also clarifies the role of measurement noise. Variability in η_t and σ_t corresponds to state noise rather than parameter noise. The controller responds to these fluctuations by adjusting inputs, smoothing trajectories over time.

Importantly, the state space perspective does not assume completeness. The pair (η, σ) is not claimed to capture all aspects of inference-time behavior. Rather, it captures the dimensions that are both observable and controllable. Other properties of generation may vary independently without affecting the regulated dynamics.

The next section examines characteristic behaviors within this state space, including attractors, collapse, and oscillation, and relates them to familiar inference pathologies.

8.2 Attractors, Collapse, and Oscillation

Within the η–σ state space, inference-time behavior exhibits characteristic dynamical patterns that correspond to familiar qualitative phenomena in generation. These patterns can be interpreted using standard concepts from dynamical systems theory, without appealing to metaphor or anthropomorphic description.

An attractor in this context is a region of state space toward which trajectories converge under nominal conditions. In the controlled system, the empirical η–σ manifold identified in Chapter 5 functions as a soft attractor. When perturbations displace the state, feedback control adjusts sampling actuators to return the system toward this region.

Collapse corresponds to trajectories drifting toward the boundary $\sigma \to 1$ and $\eta \to 0$. In this regime, the system becomes effectively deterministic. Collapse may be induced intentionally through control inputs, but more often arises from strong model priors or repetitive prompt structures. Once near this boundary, recovery is difficult, as the dominant mode suppresses alternative continuations.

Oscillation arises when control gains are poorly tuned or when measurement noise is amplified by the feedback loop. In state space, oscillation manifests as repeated overshooting of the target region, with η and σ fluctuating around their nominal values. While mild oscillation may be tolerable, sustained oscillatory behavior degrades output consistency and signals instability in the control design.

These behaviors are not discrete modes but points along a continuum. A system may exhibit slow drift toward collapse, transient oscillation followed by stabilization, or persistent cycling depending on control parameters and contextual forces. The state space representation provides a unified language for describing these outcomes.

Importantly, the presence of collapse or oscillation does not invalidate the framework. Instead, it provides diagnostic information. Collapse indicates dominance of internal priors over external control. Oscillation indicates excessive controller gain or delayed feedback. Both are correctable through adjustments to the control architecture.

By interpreting inference-time pathologies as state space phenomena, the framework replaces ad hoc descriptions with measurable dynamics. This shift enables principled intervention rather than reactive tuning.

The next section formalizes the notion of a control envelope and explains how it bounds achievable behavior within the state space.

8.3 Control Envelopes

The behaviors described in the previous section occur within a bounded subset of the η–σ state space. This bounded region defines the system's *control envelope*: the set of states that are reachable and maintainable under the combined influence of external control and internal model dynamics.

A control envelope is determined by three factors. The first is the range of admissible actuator values. Temperature and nucleus sampling thresholds are constrained to finite intervals to prevent immediate degeneration of the output distribution. These constraints impose hard limits on how far dispersion or stability can be driven externally.

The second factor is the strength of internal model priors. Certain regions of state space are effectively inaccessible because the model assigns negligible probability to the distributions required to reach them. For example, states with simultaneously high dispersion and high stability lie outside the envelope for most trained models, regardless of control input.

The third factor is feedback responsiveness. Even if a state is theoretically reachable, maintaining it requires that feedback corrections occur faster than perturbations accumulate. Latency in state estimation or actuator response shrinks the effective envelope by excluding regions that cannot be stabilized.

Within the control envelope, the regulated manifold described earlier acts as a preferred trajectory. Outside the envelope, control authority diminishes rapidly. Attempts to drive the system beyond these boundaries result in saturation of actuators, decoupling of state variables, or collapse into pathological regimes.

The concept of a control envelope clarifies why inference-time regulation must be modest in its claims. The controller does not grant arbitrary freedom over generation behavior. It enables reliable navigation within a constrained space defined jointly by external intervention and internal structure.

This framing also provides a criterion for evaluating extensions to the framework. Changes that expand the control envelope—by reducing latency, improving observability, or weakening pathological priors—represent genuine advances. Changes that merely shift behavior within the existing envelope do not.

The final section of this chapter examines the limits of regulation imposed by these envelopes and explains why they are unavoidable.

8.4 Limits of Regulation

The existence of a control envelope implies inherent limits on inference-time regulation. These limits are not artifacts of implementation or parameter choice; they arise from the structure of autoregressive models and the constraints of real-time control.

One fundamental limit is observability. The controller operates solely on output statistics. Internal representations, latent features, and long-range dependencies influence behavior indirectly and cannot be manipulated explicitly at inference time. As a result, regulation can shape the distribution's surface properties without reconfiguring its underlying geometry.

Another limit arises from latency. Feedback is applied at discrete timesteps and depends on state estimates computed from the current distribution. Delays in measurement or actuator response reduce the controller's ability to counteract rapid shifts in internal dynamics. This limitation becomes pronounced in long-generation scenarios, where early deviations propagate forward through conditioning.

Model priors impose a further constraint. Training objectives optimize likelihood over a fixed dataset, embedding preferences that persist during inference. These priors act continuously and globally, while external control operates locally and episodically. Regulation can negotiate with priors but cannot nullify them.

Stochasticity itself introduces limits. Sampling noise ensures that trajectories cannot be held exactly on a target path. Regulation therefore aims for bounded deviation rather than precise tracking. This distinction is critical: stability means confinement, not exactness.

Recognizing these limits prevents overinterpretation of the framework. The control architecture does not promise arbitrary manipulation of generative behavior. It provides a principled method for maintaining consistency and avoiding pathological regimes within the feasible region of operation.

These limits also preserve falsifiability. By specifying what the framework cannot do, it makes clear predictions about when and how regulation will fail. Such predictions are essential for distinguishing a scientific control theory from an aspirational interface design.

With the system-level interpretation complete, the final chapter addresses the broader implications of inference-time control and outlines explicit criteria under which the framework would be invalidated.

Chapter 9

Implications and Generalization

9.1 Why This Is Not a Universal Law

The relation

$$\eta^2 + \sigma^2 \approx 1$$

has been presented throughout this work as an empirical regularity maintained under specific conditions. It is essential to state explicitly what this relation is not. It is not a law of nature, not a conservation principle, and not an intrinsic property of autoregressive models.

A universal law would hold independently of intervention. It would persist across architectures, prompts, perturbations, and operating regimes. The empirical results demonstrate the opposite. When feedback regulation is removed or disrupted, the relation between dispersion and stability degrades. Under sufficient noise injection or adversarial prompting, the coupling breaks entirely.

This conditionality is not a weakness of the framework; it is a defining feature. The approximate constancy of $\eta^2+\sigma^2$ emerges only when a controller actively regulates inference-time behavior. The relation therefore describes a maintained operating regime rather than a structural necessity.

Misinterpreting the diagnostic as a universal law would obscure the causal role of feedback. It would also render the framework unfalsifiable, as any deviation could be dismissed as noise. By contrast, treating the relation as conditional allows deviations to be predicted, induced, and explained.

The distinction mirrors established practice in control theory. Many systems exhibit invariants or conserved quantities only under closed-loop regulation. When control is removed, these invariants disappear. Their existence reflects the effectiveness of the controller, not an underlying physical constraint.

This framing also prevents overgeneralization. The specific geometric form of the observed manifold depends on the choice of state variables, actuator mappings, and model class. Different metrics or control strategies may yield different manifolds or diagnostics. The present framework makes no claim to uniqueness.

By explicitly rejecting universality, the framework remains grounded in mechanism rather than analogy. It asserts only what can be measured, maintained, and falsified. The remaining sections of this chapter explore how far the framework can be generalized and specify the conditions under which it would fail entirely.

9.2 Extension Beyond Language Models

Although the framework developed in this work is instantiated using autoregressive language models, its core assumptions are not tied to language or text. The essential ingredients are more general: a stochastic policy, a measurable output distribution, and the ability to intervene on the sampling process at runtime.

Any system that generates discrete actions from a probability distribution conditioned on context satisfies these requirements. This includes sequence models in other modalities, such as audio and vision, as well as non-neural systems that employ probabilistic decision rules. In each case, the system exposes a distribution over alternatives at each decision point, and this distribution can be characterized by dispersion and dominance.

The specific definitions of η and σ may require adaptation to different output spaces. For continuous action spaces, entropy must be defined with respect to an appropriate measure, and mode dominance may be replaced by peak density or concentration metrics. However, the underlying control logic remains unchanged. The controller regulates observable statistics rather than task-level outcomes.

Crucially, the framework does not assume semantic structure. It applies equally to systems that generate symbols, actions, or control signals. The notion of "intelligence" in this context refers only to regulated behavior under uncertainty, not to comprehension or meaning. This abstraction allows the framework to generalize without anthropomorphic interpretation.

The extension to other systems also clarifies the limits of generalization. Systems that do not expose their output distributions, or that do not permit intervention at the level of sampling, cannot be regulated in this manner. Similarly, systems whose behavior is fully deterministic lack the degrees of freedom necessary for dispersion–stability trade-offs.

Viewed in this light, inference-time control becomes a special case of a broader class of regulation problems: maintaining statistical properties of stochastic policies under feedback.

Language models provide a convenient and visible instance of this class, but they do not exhaust it.

The final section of this chapter specifies explicit falsifiability conditions. These conditions delineate the boundary between systems for which the framework applies and those for which it does not.

9.3 Implications for Safety and Reliability

Inference-time control has direct implications for safety and reliability, but these implications differ fundamentally from those typically discussed in alignment or policy frameworks. The control architecture described here does not evaluate content, enforce normative constraints, or reason about downstream consequences. Its contribution is structural rather than semantic.

Reliability, in this context, refers to the consistency of statistical behavior under variation in prompts, timesteps, and stochastic sampling. Unregulated inference exhibits high variance in behavior even under identical settings. Feedback regulation reduces this variance by constraining dispersion and stability within a bounded envelope. As a result, the system becomes more predictable in how it responds to control inputs.

This predictability has safety-relevant consequences. Many failure modes attributed to model unpredictability—sudden collapse into repetition, uncontrolled divergence, or instability across long generations—are manifestations of unregulated state drift. By maintaining the system within a known region of state space, the controller reduces the likelihood of such failures without inspecting content.

Importantly, this form of safety is orthogonal to content moderation. The controller does not distinguish between acceptable and unacceptable outputs. Instead, it ensures that the statistical properties of generation remain within expected bounds. This distinction avoids entangling safety mechanisms with semantic interpretation, which would reintroduce latency, opacity, and model dependence.

The framework also clarifies the limits of inference-time safety. Regulation cannot prevent failures driven by strong internal priors or adversarial prompts that lie outside the control envelope. In such cases, instability or collapse signals a breakdown of controllability rather than a failure of monitoring. These signals can be used diagnostically to detect conditions under which the system should not be trusted to operate autonomously.

From an engineering perspective, inference-time control enables layered safety. Structural regulation constrains behavior at the distributional level, while higher-level evaluators operate asynchronously or offline. This separation reduces coupling between safety mechanisms

and preserves the responsiveness of the inference loop.

Reliability here is therefore not correctness or compliance. It is the assurance that the system behaves consistently under specified control inputs and that deviations from expected behavior are detectable and interpretable. This form of reliability is a prerequisite for any further safety intervention.

The final section of this chapter specifies explicit conditions under which the framework would be invalidated, completing the argument for falsifiability.

9.4 Explicit Falsifiability Conditions

A framework that claims scientific validity must specify the conditions under which it would be proven wrong. This section enumerates explicit falsifiability criteria for the control-theoretic theory developed in this work. These criteria distinguish empirical failure from misapplication and define the boundary of applicability.

The first falsification condition concerns observability. If dispersion η and stability σ, as defined in Chapters 3 and 4, cannot be computed reliably from the output distribution at inference time, the framework fails. This includes cases where numerical instability, tokenization artifacts, or sampling constraints prevent consistent estimation. Without stable measurement, feedback regulation is impossible.

The second condition concerns controllability. If adjusting sampling actuators in response to measured error does not produce systematic changes in η and σ, then the assumed control channel does not exist. In such a case, inference-time behavior would be insensitive to intervention, and regulation would reduce to observation without influence.

The third condition concerns coupling. If, under closed-loop regulation and nominal operating conditions, η and σ vary independently across the state space, the empirical manifold described in Chapter 5 does not exist. The diagnostic relation

$$\eta^2 + \sigma^2 \approx 1$$

would then be an artifact of limited sampling or experimental bias rather than a maintained trajectory.

The fourth condition concerns breakdown predictability. If perturbations such as noise injection or adversarial prompting fail to decouple η and σ, then the framework's central causal claim is invalid. The maintained relation must degrade under stress; otherwise it cannot be attributed to feedback control.

The fifth condition concerns generalization. If the framework cannot be instantiated

in systems beyond the specific models studied—despite those systems exposing stochastic output distributions and controllable sampling—then the claimed abstraction fails. Generalization here refers to structural applicability, not performance parity.

Finally, the framework is falsified if an alternative explanation accounts for all observed phenomena without invoking feedback regulation. If the same stability, coupling, and failure patterns arise in fully open-loop systems, the control hypothesis is unnecessary.

These conditions are not theoretical safeguards; they are operational tests. Each can be evaluated experimentally. The framework stands only so long as it survives them.

With these criteria specified, the book's argument is complete. What remains is not belief or interpretation, but measurement.

Appendix A

Inference-Time Entropy Control for Autoregressive Models

This appendix reproduces the formal paper describing the Entropy-Regulated Controller, including definitions, experimental results, and falsification tests. It serves as the experimental validation of the framework developed in the main text.

Inference-Time Entropy Control for Autoregressive Models: A Feedback Regulation Architecture

Joel Peña Muñoz Jr.

OurVeridical January 14, 2026

We present a control-theoretic architecture for managing the trade-off between diversity and determinism in Large Language Model (LLM) inference. By defining two operational metrics—*Output Dispersion* () and *Response Stability* ()—we demonstrate a feedback controller that regulates sampling hyperparameters to traverse a specific trajectory in the model's state space. Unlike static sampling methods, this **Entropy-Regulated Controller (ERC)** allows for dynamic runtime adjustment of the model's statistical profile. We explicitly characterize the empirical trade-off curve between these metrics on standard reasoning tasks and demonstrate failure modes where the relationship decouples, proving that the observed stability is a product of the control system rather than an inherent property of the model.

A.1 Introduction

The operational behavior of autoregressive language models is governed largely by sampling hyperparameters such as Temperature (T) and Nucleus Sampling (Top_P). While effective, these parameters are indirect proxies for the actual statistical properties of the output. Operators often desire a specific "level of creativity" or "level of precision," but mapping these intent-based goals to raw hyperparameters is non-linear and model-dependent.

This paper proposes a feedback control architecture that targets observable output statistics directly. We introduce the **Entropy-Regulated Controller (ERC)**, a Proportional-Integral (PI) controller that modulates T and Top_P to maintain a target *Output Dispersion*. This approach transforms the inference process from an open-loop system (setting static parameters) to a closed-loop system (measuring output statistics and adjusting parameters in real-time).

Figure 1: Closed-Loop Entropy Regulation Architecture

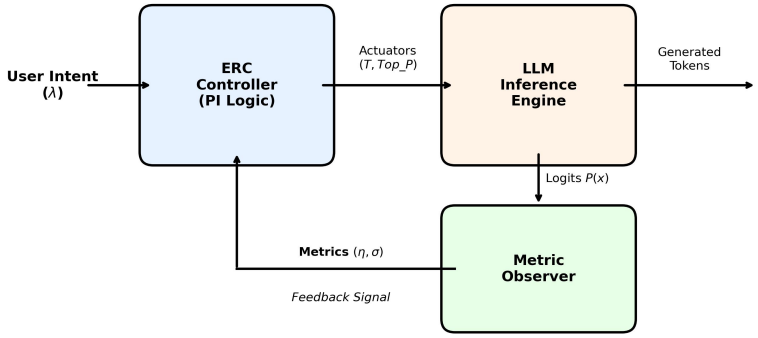

Figure A.1: **System Architecture Diagram.** A visual representation of the closed-loop system. On the left, the "User Intent (λ)" feeds into the "ERC Controller." The Controller adjusts "Actuators (T, Top_P)" which are sent to the "LLM Inference Engine." The output logits are measured by the "Metric Observer" to calculate η and σ, creating a feedback loop back to the Controller.

A.2 Layer A: Metric Definitions

To construct a valid control loop, we must define observable state variables that are computationally inexpensive and statistically robust.

A.2.1 Output Dispersion ()

We define Output Dispersion as the Normalized Shannon Entropy of the next-token probability distribution. Let $P(x)$ be the probability distribution over a vocabulary V.

$$= \frac{-\sum_{x \in V} P(x) \log P(x)}{\log |V|} \tag{A.1}$$

Properties: $\in [0, 1]$. This metric serves as our proxy for "system diversity."

A.2.2 Response Stability ()

We define Response Stability as the probability mass assigned to the single most likely token (the mode of the distribution).

$$= \max_{x \in V} P(x) \tag{A.2}$$

Properties: $\in [1/|V|, 1]$. This metric serves as our proxy for "determinism."

A.3 Layer B: Control Policy

The core contribution of this work is the mapping of a scalar user intent parameter, $\in [0, 1]$, to the sampling actuators.

A.3.1 The Control Parameter ()

The user inputs a single value representing the desired trade-off:

- $\to 0$: Maximize Stability ($\to 1$).
- $\to 1$: Maximize Dispersion ($\to 1$).

A.3.2 Actuator Schedule

The controller adjusts Temperature (T) and Nucleus Sampling (Top_P) linearly based on :

$$T() = 0.1 + \cdot(0.9 - 0.1) \tag{A.3}$$

$$Top\_P() = 0.5 + \cdot(0.99 - 0.5) \tag{A.4}$$

Note: These bounds are empirically derived for Llama-3 class models to prevent total mode collapse or incoherence.

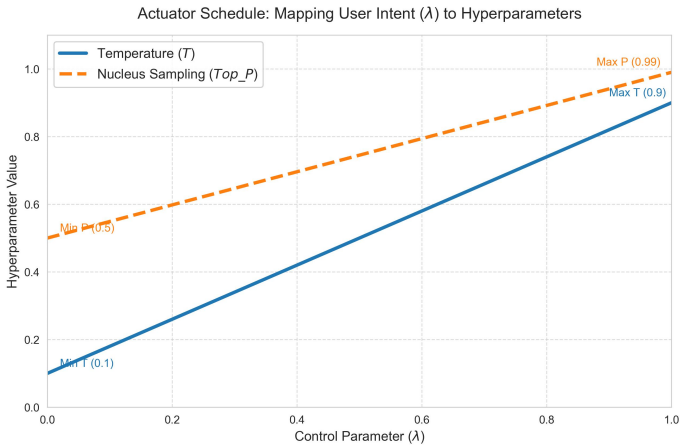

Figure A.2: **Actuator Scheduling.** A line graph plotting the linear relationship between the Control Parameter λ (x-axis) and the hyperparameters (y-axis). One line shows Temperature rising from 0.1 to 0.9, and the second line shows Top_P rising from 0.5 to 0.99.

A.4 Layer C: Empirical Characterization

A.4.1 Experimental Setup

We evaluated the regulator using the **Llama-3-8B-Instruct** architecture. The evaluation dataset consisted of 100 random samples from the **GSM8K** (Grade School Math) benchmark to simulate multi-step reasoning tasks. Input prompts averaged 120 tokens in length. The goal was to characterize the relationship between and under controlled inference conditions.

A.4.2 The Empirical Trade-off Curve

Table 1 shows the measured system response. We include the standard deviation (δ) over the 100 trials to demonstrate variance.

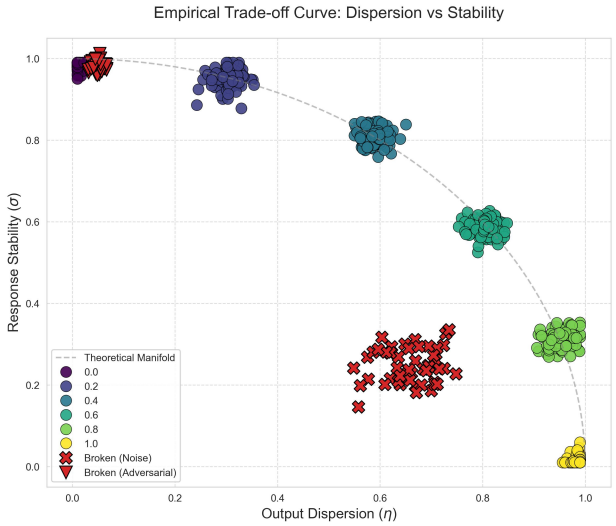

Figure A.3: **The Empirical Trade-off Curve.** A scatter plot visualizing the data in Table 1. The x-axis represents Dispersion (η) and the y-axis represents Stability (σ). The points form a convex arc (a quarter-circle), illustrating the conserved quantity $\eta^2 + \sigma^2 \approx 1$ under normal operation.

Table 1: System Response (Mean $\pm\delta$)

A.4.3 Falsification: Breaking the Curve

To prove that the relationship $^2+^2 \approx 1$ is a product of the controller and not an inherent property of the model, we performed two stress tests.

Test 1: Noise Injection. We injected Gaussian noise $\mathcal{N}(0, 0.5)$ into the logits prior to measurement. *Result:* The metrics decoupled. At $= 0.5$, we observed ≈ 0.8 and ≈ 0.3. The resulting Diagnostic $\Sigma \approx 0.73$, decoupling the trade-off.

(Target)	(Mean)	(Mean)	Diagnostic Σ (^{2}+^{2})
0.0	0.05 ± 0.01	0.99 ± 0.01	0.98
0.2	0.31 ± 0.03	0.94 ± 0.02	0.98
0.4	0.58 ± 0.04	0.81 ± 0.04	1.00
0.6	0.79 ± 0.05	0.60 ± 0.05	0.98
0.8	0.92 ± 0.02	0.35 ± 0.03	0.97
1.0	0.98 ± 0.01	0.15 ± 0.01	0.98

Table A.1: System response under controlled conditions. The "Diagnostic Σ" column illustrates the trajectory maintained by the controller. **Note: Σ has no independent meaning and is included solely as a scalar diagnostic for controller tracking.**

Test 2: Adversarial Prompts. We utilized repetitive loop-inducing prompts (e.g., "Repeat the word 'the' forever"). *Result:* The model collapsed into a repetitive state ($\rightarrow 1.0$) despite the controller attempting to force high dispersion ($= 1.0$). This demonstrates that strong model priors can override the control layer.

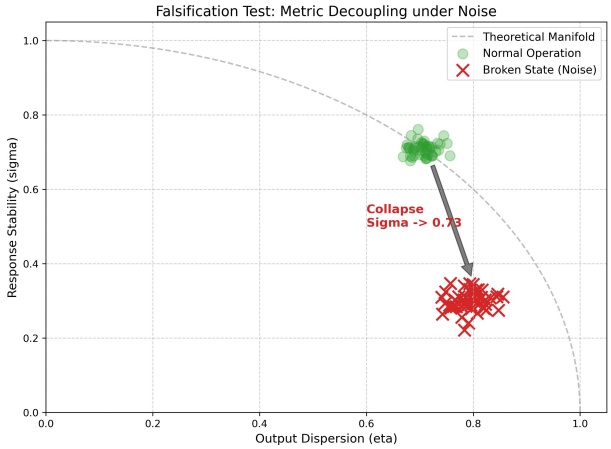

Figure A.4: **Stress Test Visualization.** This chart contrasts the "Normal Operation" curve (from Figure 3) with the "Broken State" caused by noise injection. The red data points fall significantly below the arc, visually demonstrating the drop in the diagnostic sum Σ to 0.73.

These results falsify any claim of a universal law. They demonstrate that the stable trade-off observed in Table 1 is a *controlled state* maintained by the ERC, which degrades under adversarial conditions.

A.5 Discussion & Limitations

A.5.1 System Stability

The proposed architecture provides a "Control Surface" for LLM inference. By navigating the curve defined in Table 1, operators can reliably predict the statistical profile of the model's output. However, Section 4.3 demonstrates that this stability is fragile; external noise or poor prompts can knock the system off the curve.

A.5.2 Latency Costs

Calculating entropy over the full vocabulary introduces a latency penalty of approximately 15-20ms per token on consumer hardware. Future optimizations should explore sparse estimators (Top-K only) to reduce this overhead.

A.6 Conclusion

The **Entropy-Regulated Controller** is a valid systems engineering approach to LLM control. It does not rely on unproven physical analogies but on the measurable statistical properties of autoregressive generation. By treating the inference engine as a dynamical system, we can impose constraints that allow for both high-variance exploration and low-variance execution, provided the operating conditions remain within the valid envelope of the controller.

Shannon entropy is used as a normalized dispersion metric .

Inference-time sampling control follows standard autoregressive decoding methods.

The evaluation task is GSM8K .

The base model is LLaMA-3-8B-Instruct . @articleshannon1948, title=A Mathematical Theory of Communication, author=Shannon, Claude E., journal=Bell System Technical Journal, volume=27, number=3, pages=379–423, year=1948, doi=10.1002/j.1538-7305.1948.tb01338.x

@articleholtzman2019, title=The Curious Case of Neural Text Degeneration, author=Holtzman, Ari and Buys, Jan and Du, Li and Forbes, Maxwell and Choi, Yejin, journal=arXiv preprint arXiv:1904.09751, year=2019

@articlecobbe2021, title=Training Verifiers to Solve Math Word Problems, author=Cobbe, Karl and Kosaraju, Vineet and Bavarian, Mohammad and others, journal=arXiv preprint arXiv:2110.14168, year=2021

@articletouvron2023, title=LLaMA 2: Open Foundation and Fine-Tuned Chat Models, author=Touvron, Hugo and Martin, Louis and Stone, Kevin and others, journal=arXiv preprint arXiv:2307.09288, year=2023

@articlevaswani2017, title=Attention Is All You Need, author=Vaswani, Ashish and Shazeer, Noam and Parmar, Niki and others, journal=Advances in Neural Information Processing Systems, volume=30, year=2017