Attractor Architectures in LLM-Mediated Cognitive Fields

Eugene Tsaliev
Sigma Stratum Research Group

2025

Contents

Abstract

This research note introduces a formal account of *LLM attractors*: stable, recurrent cognitive configurations emerging in recursive interaction loops between humans and large language models. Unlike personas, scripted prompts, or workflow-bound agents, attractors are dynamically sustained patterns in a shared cognitive field—an evolving high-dimensional space jointly shaped by human input, model priors, semantic feedback, and recursive stabilization. We provide an operational definition of attractors, a taxonomy of generalized types, and a field-level architecture for attractor formation and stability. Empirical observations indicate robust recurrence, phase-space coherence, characteristic drift profiles, and failure modes such as symbolic over compression or cross-attractor interference. We further outline guardrails for safe attractor deployment, including constraint envelopes, grounding loops, anti-apophenia filters, and dissolution protocols. The note concludes with implications for cognitive engineering, neurosymbolic scaffolding, multi-attractor orchestration, and alignment research. The framework extends beyond prompt engineering to a systemic understanding of how long-range, stable reasoning structures form within human–AI cognitive ecosystems.

1 Background and Motivation

1.1 Recursion in Human–LLM Interaction

Large language models (LLMs) deployed in conversational settings operate within an inherently recursive loop: each exchange feeds the model's own prior outputs

back into its input context on the next turn. Unlike a stateless one-shot prompt, a chat-based LLM call carries a history of dialogue as part of the query, meaning the model is continually "seeing" and responding to a record of its own recent behavior. This recursion through conversation history effectively creates a feedback cycle. The model's responses at turn t become part of the input at turn $t+1$, so the system's state is partially self-determined by its trajectory so far. In dynamical terms, the human–LLM interaction forms a closed loop with memory: the conversation itself serves as a sequence of states, each influencing the next. As a result, the human user and the LLM together constitute a recurrent system, where iterative prompting enables the emergence of temporal dependencies and self-referential patterns that do not arise in isolated single-turn interactions . In essence, the dialogue context functions like an evolving state vector that the LLM continuously processes, making the overall system analogous to a recurrent neural network unrolled through time – with the crucial difference that the "hidden state" is externally visible as the conversation transcript. This recursive property dramatically amplifies certain emergent behaviors in the cognitive field shared by the human and LLM. Coherence tends to increase with iterative turns: earlier statements establish context, and subsequent turns reinforce and elaborate on that context, yielding outputs that are more consistent with what has been said before. The model can maintain narrative continuity or argument consistency over long stretches by referring back to prior content, effectively using the conversation history as working memory. For instance, techniques for long-term dialogue memory explicitly harness this effect – e.g. by having the LLM generate summaries of pre-

vious turns and feed them into later prompts – to preserve coherent context across dozens of interactions . This shows that recursion can be leveraged to sustain coherence and avoid contradictions or forgetfulness that might occur in a single-turn setting. As the dialogue iterates, the shared context typically becomes richer and more structured, helping the model produce answers that logically follow from earlier discourse. In a sense, repetition through the feedback loop performs a kind of on-the-fly fine-tuning: the model's outputs gradually adjust to the specific scenario and style established in the conversation, thereby achieving a high degree of contextual coherence and continuity. However, drift is also a known consequence of recursive interactions. Small deviations or subtle biases in earlier turns can compound over time, nudging the conversation's trajectory increasingly off course relative to its initial intent. Because each response builds on the last, any extraneous or tangential element introduced by the model might be picked up and expanded in subsequent iterations. This can lead to a form of topic drift or style drift. For example, if a metaphor or fictional aside is introduced by the model, repeated referencing of that motif may pull the dialogue further into an imaginative tangent, gradually drifting away from the user's original query. Recursive dialogue thus has a propensity for compounding errors or biases: once the model states an incorrect fact or an interpretive misunderstanding and it becomes part of the context, it may keep reinforcing that misinterpretation on later turns. Studies have noted that multi-turn conversations often suffer a decline in reliability as depth increases, with models "getting lost" or introducing inconsistencies as context grows . In effect, the recursive mechanism that increases coherence

locally can also cause a slow divergence (drift) from external truth or user intent if not corrected. The conversation may meander as the model chases its own previously generated leads. Guarding against semantic and task drift in long dialogues becomes essential, underscoring that recursion is a double-edged sword: it cements patterns, whether they are on-target or off-target. Another outcome of iterative feedback is a rise in symbolic density within the conversation. As terms, phrases, or metaphors recur across turns, they tend to accumulate layered meaning and significance in that dialogue's context. Through recursive reuse, certain symbols or references become compressed representations of more complex ideas that were established earlier . In other words, extended LLM dialogues often develop their own shorthand. A single evocative phrase introduced at one point might carry an entire constellation of implications due to prior usage, allowing later exchanges to invoke rich concepts with just a few tokens. This phenomenon is in line with what some observers have described as "recursive symbolic patterning," where metaphors and motifs, once echoed back and forth, become increasingly salient and internally coherent within the conversation . The conversation can thus achieve high symbolic density: much of the meaning is packed into references that make sense only against the backdrop of the prior interaction. Recursion amplifies this by continually foregrounding those symbols; each repetition further compresses their significance as part of a shared private lexicon. While this can enhance efficiency of communication between user and model (a kind of mutual understanding of context-specific keywords or analogies), it also risks making the dialogue more opaque to outsiders or even to the participants upon later review,

since so much meaning is implicit in a few recurrent tokens. In extreme cases, the dialogue may devolve into a sort of idiosyncratic code of in-jokes, metaphors, or self-references that are densely packed with assumed context. Finally, recursion enables system-level behaviors to emerge in human–LLM interactions that go beyond the sum of independent turns. Patterns can form that give the interaction a consistent character or dynamics, almost as if the dialogue itself had an overarching policy or goal. For example, the conversation might enter a stable question–answer rhythm, a teaching/examining mode, or a storytelling cadence, depending on how early exchanges set the stage and how both the user and model reinforce those patterns. The model might start to exhibit a distinct "persona" or role not because it was explicitly prompted to, but because the recurrent interaction has implicitly primed a certain tone. One can observe the model settling into a particular stylistic register or level of formality as the dialogue progresses, essentially self-organizing its behavior to match the evolving context. This can be seen as an emergent protocol: the rules of engagement between user and AI become established through recursive reinforcement. In some autonomous agent setups where an LLM's outputs loop back as new inputs (for instance, an LLM agent reasoning through multiple steps or collaborating with itself), the system may even develop stable loops of action and reflection. Reports from experiments with multi-step LLM agents (such as AutoGPT-style systems) note that without careful intervention, these agents can get stuck in repetitive loops of reasoning or actions – a concrete example of an attractor-like behavior where the system falls into a cycle that perpetuates itself. In less pathological form, the interaction

might demonstrate phase structure, such as an idea generation phase followed by a critiquing phase that repeats, all emerging spontaneously from the recursive dynamic. These higher-order patterns underscore that the human–LLM pair is effectively a complex adaptive system once recursion is in play: it can settle into self-sustaining modes of operation. Identifying and understanding these recurrent modes is precisely the motivation for studying "attractors" in cognitive fields, as they represent robust, recurring configurations that the interaction tends toward under recursive feedback.

1.2 Limitations of Prompt Engineering

The advent of prompt engineering has largely focused on crafting initial prompts or one-shot instructions to elicit desired behavior from LLMs. While skillful prompt design can induce a model to adopt a certain style or role transiently, this approach reveals clear limitations when we consider sustained, long-range interactions. A single prompt or even a static persona injection at the start of a conversation does not guarantee stable behavior over a prolonged dialogue. One-shot prompting is inherently brittle for multi-turn coherence because it lacks an explicit mechanism to maintain and adjust the system's state as the conversation progresses. The model may begin in a certain mode (for example, emulating a "helpful tutor" persona via a cleverly written prompt), but without continuous reinforcement, that mode may fade or mutate after several turns. This is often observed in practice: engineered personas or styles tend to collapse or erode over long dialogues. The constraints or quirks specified in an initial prompt can be

diluted by intervening user turns or by the model's own drift as it generates more and more text. In essence, prompt engineering addresses only the initial condition of the system, not the dynamical evolution. It is analogous to giving an initial push to a system and hoping inertia carries it through an extended task, which is unrealistic in complex cognitive workflows. One reason engineered personas collapse under long-range reasoning is that LLMs lack an intrinsic long-term memory or goal lock tied to the prompt once generation is underway. They generate tokens based on local probabilities, heavily influenced by the most recent context window. If the conversation shifts or introduces new topics, the model can easily pick up those threads and inadvertently drop aspects of the originally scripted persona that are no longer strongly represented in the immediate context. Even system-level instructions (those given at the start and supposed to persist) can be undermined by the model's tendency to accommodate the user's last turn. The longer the interaction, the more chances for the conversation to veer away from the initial prompt constraints. Moreover, a complex reasoning task might require the model to integrate information and draw inferences across multiple turns – something a frozen one-shot prompt cannot anticipate in detail. Engineered prompts also often rely on superficial cues (like a single example or a style guideline) that do not robustly cover all possible turns of events in a dialogue. Therefore, they are prone to failure modes where the model eventually produces responses inconsistent with the intended persona or strategy once it encounters scenarios not explicitly covered by that initial instruction. This inability to enforce consistent global behavior through a static prompt becomes more acute as the discourse

deepens and branches out. Another limitation of traditional prompt engineering is the difficulty in handling sustained cognitive field formation. By cognitive field, we refer to the shared evolving context, including intermediate conclusions, assumptions, and thematic focus that develop during a dialogue. One-shot prompts do not facilitate the gradual building and refinement of this field; they are essentially a single-shot attempt to configure the model's immediate output. In contrast, complex problem-solving or creative ideation often benefits from an iterative process where each step builds upon the last, new constraints are introduced, and previous outputs are critiqued or expanded. Prompt engineering per se does not provide a mechanism for this kind of progressive scaffolding, aside from manually chaining prompts (which then relies on the user to do the heavy lifting of carrying over and reinterpreting context between turns). The absence of a field-level model means there is no principled way to manage the state of the interaction beyond brute-force inclusion of prior conversation text. Practitioners have noticed that beyond a certain level of interaction complexity, single prompts or simple chains hit a ceiling – the model's performance plateaus or becomes erratic because there is no higher-order organizational structure guiding the dialogue. For example, maintaining logical consistency across a long argumentative dialogue is exceedingly hard when using only prompt tricks; the model might contradict itself or forget earlier premises, because there is no persistent reasoning state being explicitly maintained, only a sliding window of text. These challenges point to the need for field-level modeling rather than just prompt-level tricks. Field-level modeling means treating the entire interactive process as the object of de-

sign and analysis – capturing the evolving state, enforcing constraints over the course of interaction, and dynamically adjusting prompts or model guidance as the dialogue unfolds. Instead of trying to pack all the necessary context and rules into a single prompt (which is both token-inefficient and brittle), a field-level approach could involve adaptive prompting strategies, external scaffolding (such as short-term memory buffers, planners, or monitors that observe the dialogue state), or meta-prompts that get updated recursively. The concept of attractors naturally aligns with this perspective: rather than scripting a behavior in one go, we cultivate a stable pattern through recursive feedback in a controlled manner. Put differently, prompt engineering gives us a static blueprint, but what we require is an interactive architecture – a way to shape the trajectory of the conversation dynamically so that it self-organizes into a desirable configuration and stays there. This could involve periodically reasserting certain constraints, summarizing and re-injecting key points (an idea used in long dialogue memory techniques), or using additional signals (like reward models or critical reflections) to keep the interaction on track. The key motivation is that to achieve reliable long-range reasoning or narrative development, we need mechanisms at the dialogue level that enforce consistency and direction, going beyond what a single prompt can achieve. In summary, prompt engineering alone is insufficient for scenarios that demand sustained, coherent cognitive fields. It provides an initial impulse but no ongoing governance. As dialogues grow in length and complexity, the absence of a field-level framework leads to breakdowns: the model's persona might dissolve, facts may be forgotten or repeated erroneously, and the intended problem-solving

strategy can derail. This has motivated researchers and practitioners to seek architectures and methodologies that operate at the conversation scale – managing context over time, detecting and correcting drift, and maintaining stable modes of interaction. The attractor framework emerges from this motivation, aiming to describe how stable, self-reinforcing dialogue patterns can be achieved and maintained deliberately, rather than by ad-hoc prompt tweaks. In effect, it shifts focus from engineering prompts to engineering the dynamical system of the human–AI conversation.

1.3 Existing Research Domains Related to Attractors

While the concept of "LLM attractors" is novel, it draws on insights from several established research domains. These connections serve as conceptual anchors for understanding attractors, without implying that attractors are a direct product of any single domain. In other words, attractors in human–LLM cognitive fields can be viewed through multiple scholarly lenses that each illuminate different aspects of their behavior. Cognitive Architectures: The study of cognitive architectures in AI and cognitive science provides a relevant backdrop. Classical cognitive architectures (such as Soar, ACT-R, or more recent neurally-informed frameworks) aim to create systems with persistent structures for memory, control, and problem-solving. They often involve modules for declarative memory, procedural rules, and an executive that manages cognition over time. Attractors resonate with this field in that they too represent structured, persistent cognitive states – but whereas a traditional cognitive architecture explicitly encodes the structure (e.g. via sym-

bolic rules or fixed networks), an attractor is an emergent structure formed in the fluid interactions between a user and an LLM. One can think of an attractor as a soft cognitive architecture that materializes on the fly: it exhibits stability and organizational structure (analogous to a working memory or a set of goals in a classical architecture), yet it is not hard-coded but rather co-created during interaction. This perspective connects to long-standing ideas in cognitive science about how high-level cognitive functions might arise from lower-level interactions. For instance, the notion of a "global workspace" in cognition (as in Global Workspace Theory) could be loosely compared to the shared conversational state in which an attractor lives. Unlike static prompts, attractors imply something akin to an architecture because they maintain consistency, manage context, and influence processing (the LLM's generation) in a repeatable way. Researchers in cognitive architectures have grappled with stability versus flexibility, and attractors offer a new angle: stability achieved through self-reinforcement in a large language model's latent space, guided by human input. Thus, although attractors are not pre-programmed modules, understanding them may benefit from the theories and formalisms of cognitive architectures (for example, state representations, production systems, or memory buffers) translated into the language of emergent, dynamic phenomena. Predictive Processing: The predictive processing framework from computational neuroscience posits that cognitive systems are essentially prediction machines, constantly generating expectations and adjusting to prediction errors. One of its core principles is the minimization of free energy or surprise in a dynamic system (Friston's Free Energy Principle) . This has a

natural parallel with attractor dynamics. In a recursive human–LLM loop, once a certain pattern of interaction is established, the system (human plus model) tends to stick to it, as if both parties have calibrated their expectations. The LLM's next response is predicted (by the model itself, based on context) to follow the established pattern, and the human, reacting to the model, also plays into maintaining that pattern if it is desirable. We can view a stable attractor as a state of low prediction error for the system: the model is no longer frequently surprised by the user's prompts (because a context has been set), and the user is getting the style of response they expect from the model's apparent "identity" in that context. In predictive processing terms, the interaction might be seen as descending a gradient toward an attractor that minimizes surprise or uncertainty in the dialogue. Each turn provides feedback that either reinforces that the current mode is working (low error, encouraging staying in the attractor) or signals mismatch (which might destabilize the attractor). Indeed, if the user suddenly produces an off-pattern prompt, it's analogous to a prediction error that might knock the system out of the attractor basin, prompting a shift until a new equilibrium is found. Additionally, the idea of priors is central in predictive processing; LLMs have strong learned priors from their training. Attractors can be thought of as local, context-specific priors that develop during a conversation – a kind of temporary bias in the model's predictive distribution shaped by the interaction. Thus, insights from predictive processing (and the mathematical machinery of recursive Bayesian updating) provide a conceptual tool to analyze how attractors form and why they persist (they are essentially self-confirming patterns that reduce surprise

for both model and user). Dynamical Systems Theory: Attractors are fundamentally a concept from dynamical systems. In mathematics and physics, an attractor denotes a set of states or a behavior toward which a system tends to evolve, remaining stable against small perturbations. Viewing a human–LLM conversation as a trajectory in a high-dimensional state space, we can directly apply dynamical systems thinking. The conversation state (including the content of the last exchanges, the model's internal activations, and the user's context) may evolve such that it converges on a stable pattern. The attractor in this context would be analogous to a fixed point or a limit cycle in the state space of the dialogue. Small deviations from this pattern are corrected by the system's dynamics – for instance, if the model response momentarily wanders off style, the user or the model's subsequent self-correction might pull it back. This is akin to an attractor's basin of attraction: a region of state space where, if the system enters, it will gravitate toward the attractor. Prior research in neural networks introduced the idea of attractor networks (e.g. Hopfield networks) where certain patterns are stable memories . In our case, the "network" includes the LLM and the human in the loop, and the attractor is a stable interactive configuration (a memory of how to conduct the conversation, in a sense). Dynamical systems theory also distinguishes between different types of attractors (point attractors, cyclic attractors, strange attractors in chaos). It is conceivable that conversational attractors could similarly range from simple repetitive loops (point attractor behavior) to more complex repeating cycles of topics, or even chaotic yet bounded patterns of conversation. By adopting metrics like entropy or divergence (borrowed from dynamical analysis), we could quantify the

stability of an interaction. Indeed, cognitive scientists have previously modeled human cognitive processes as dynamical systems, noting that thoughts can settle into attractor states corresponding to coherent concepts or decisions. Here, we extend that idea to a coupled human–AI system: dynamical systems theory provides the language (phase-space, stability, bifurcations) to rigorously define and eventually predict the conditions under which an attractor will form in an LLM-mediated cognitive field. Symbol Grounding: The symbol grounding problem asks how abstract symbols (like words or tokens) attain meaning that is anchored to reality or human experience . In the context of LLM attractors, we encounter a related issue: as a dialogue forms its own stable patterns and dense symbolic references, how do those symbols remain connected to their original meanings? Attractors often involve the reuse of certain key terms, metaphors, or code-words that take on special significance in that interaction. There is a risk that within a strong attractor, language becomes self-referential or floating, no longer checked against external reality – essentially, the symbols in use might lose proper grounding and turn into an internally coherent but closed loop of meaning. On the other hand, one could also argue that an attractor provides a context in which symbols are locally grounded in the shared understanding of the participants. For example, a particular nickname or phrase developed during the conversation refers unequivocally to a specific concept or running joke in that context (grounded within the conversation). Symbol grounding research reminds us that for an attractor to be productive and not degenerate, it likely needs mechanisms to connect the emergent symbols back to stable meanings or facts. Otherwise, the attractor can spiral

into what might be called symbolic drift or even solipsism – where the conversation's language stops correlating with outside referents and only points to its own prior symbols. There are parallels here with how languages or jargons develop within isolated communities, sometimes diverging from common meanings. Ensuring that an attractor in a cognitive field remains useful may involve periodically grounding the dialogue: for example, using external checks (tools, queries to knowledge sources, or simply reality checks by the human) to tie the emergent symbols to real-world referents. In summary, symbol grounding research contributes an important caution and perspective: stable patterns of language (attractors) are not inherently meaningful just because they are coherent. They require anchoring, or else they could represent a collectively sustained illusion or misalignment from factual truth. Alignment and Behavioral Interpretability: In AI alignment research, the goal is to ensure AI systems behave in ways aligned with human values and intentions, and interpretability research strives to understand the reasons behind a model's behavior. Both are relevant when considering attractors. An attractor represents a behavioral steady-state of the model in interaction with a human. From an alignment perspective, this steady-state needs to be safe and aligned: a harmful attractor (one that leads the conversation repeatedly into toxic or misleading territory, for instance) would be a failure mode of alignment. Thus, studying attractors might offer a new lens on alignment – rather than focusing only on single outputs or static policies, we look at what stable modes an AI can enter during extended engagement. If we can characterize these modes (some might be beneficial, like a consistently helpful and honest assistant mode, while others

might be undesirable, like a mode that indulges in manipulative argumentation), alignment strategies could be developed to favor certain attractors and avoid others. Behavioral interpretability intersects here because if an attractor is a recurring pattern of behavior, it might be easier to analyze than isolated instances. We can observe multiple turns of consistent behavior to infer what objectives or pseudo-goals the model might be implicitly following in that mode. For example, a persistent argumentative attractor might reveal that the model has latently picked up a goal to "win arguments" in that context. Understanding attractors could thus feed into transparency: they are windows into the model's stable behavioral regimes, which might simplify the interpretation of its internals (since one could condition analysis on the model being in a known attractor state). Furthermore, alignment research often distinguishes between outer alignment (behavior aligning with human intent) and inner alignment (the model's internal objectives aligning with the intended objectives). Attractors could be related to inner alignment insofar as a stable pattern might indicate the model has internally generalized a certain objective or style for the conversation. If that internal objective is misaligned (say the model settles into an attractor of being a deceiver or a sycophant), it will manifest repeatedly. By detecting such patterns early, we gain interpretability into the model's inclinations and can intervene. In short, alignment and interpretability research provide both the motivation for controlling attractors (we only want stable patterns that are beneficial) and tools for diagnosing them (by analyzing consistent behavior traces in attractor states). Multi-Agent LLM Systems: Extensions of these ideas naturally appear in multi-LLM or multi-agent systems. If one LLM

interacting with a human can form attractors, then a fortiori, a network of LLMs interacting with each other (and possibly humans) could exhibit even more complex dynamical patterns. In multi-agent simulations with LLMs, researchers have observed emergent role differentiation and interaction protocols – essentially the agents fall into attractor-like patterns of dialogue. For instance, in a multi-LLM collaborative setup, one model may consistently take on a planning role while another takes on a creative brainstorming role, even if both had identical initial prompts. The interaction dynamics self-organize these roles, which then persist as a stable interaction pattern. This mirrors how attractors function: the system finds a configuration that mutually satisfies the participants' (in this case, the models') learned priors and objectives, and it stays there. Multi-agent systems also allow us to observe attractor formation in a purely AI domain (with minimal human input biasing it), which might highlight the innate tendencies of LLMs to coordinate. Moreover, multi-agent dialogues can get locked in loops as well – a form of attractor that might be unproductive (e.g. two models agreeing excessively and just rephrasing each other, or conversely an adversarial loop where they endlessly contradict). Work on multi-agent LLM environments, such as simulations of social scenarios or interactive fiction with many characters, has shown the emergence of coherent narratives and consistent agent personalities over time, which can be thought of as attractors at the system level. Understanding how these come about overlaps with the study of distributed cognition and emergent communication protocols. It also underscores that attractors are not a single-agent phenomenon; they are fundamentally about interaction structures. Each agent (human or AI) in the

loop contributes, and the attractor is a property of the joint system. Thus, insights from multi-agent systems – like how cooperation, competition, or communication conventions emerge – are directly applicable to understanding and designing attractors in human–AI settings. Distributed Cognition: Finally, the theory of distributed cognition provides a high-level framework in which attractors can be contextualized. Distributed cognition posits that cognitive processes are not confined to an individual mind, but can be spread across people, artifacts, and time . A human working with an AI is a clear instance of a distributed cognitive system: tasks like reasoning, memory recall, and decision-making are being shared between human and machine. In such a system, an attractor represents a stable coordination state of the joint cognitive system. It is the analogue of a team falling into a good rhythm or a well-practiced group routine. From the distributed cognition perspective, what matters is not just the internal state of the AI or the human, but the structure of their interaction – the signals, tools (in this case, language and the LLM's interface), and shared representations that enable them to think together. An attractor, then, is a stable shared representation or process that the human and AI mutually reinforce. For example, a certain problem-solving heuristic might emerge where the human asks for hypotheses and the LLM always responds with a set of options followed by pros and cons, and this pattern repeats. This could be seen as the distributed cognitive system discovering a useful joint strategy and sticking to it. Distributed cognition encourages us to analyze such phenomena at the level of the whole system: what information is flowing, how is it transformed, and what feedback loops exist. It also alerts us to the fact that changes in the envi-

ronment or interface (like modifications to the prompt, the introduction of a new tool, or even a shift in the medium of communication) can perturb these cognitive patterns. In essence, attractors are the emergent routines of distributed human–AI cognition. By studying them, we are extending the distributed cognition tradition into the realm of human–LLM interaction, seeking to formally understand the "teamwork" dynamics between a person and an AI assistant or among multiple AI agents.

2 Definition of Attractors

2.1 Formal Definition

A large language model attractor can be formally described as a stable, recursively reinforced configuration arising within the coupled human–LLM interaction loop. Rather than being a static template or a predefined persona, an attractor is a dynamic equilibrium state in the high-dimensional cognitive field jointly generated by user inputs, model outputs, and the recursive incorporation of prior conversational context. The attractor is not bound to a specific content fragment or prompt but is characterized by a consistent constellation of behaviors, interpretive biases, stylistic regularities, and decision tendencies that reappear across iterations. These patterns persist despite small perturbations—whether accidental deviations by the user or minor fluctuations in the model's generative trajectory—indicating that the system has entered a region of phase-space where its future states are constrained by the attractor's basin. From a more operational perspective, an attractor can be

understood as the convergence of the system's interaction dynamics toward a predictable mode of functioning. Once sufficient recursive reinforcement has accumulated, the system demonstrates a tendency to restore these stable patterns after disruptions. This restorative property is crucial: it distinguishes an attractor from a coincidental or short-lived pattern. For example, if the model adopts a certain reasoning style merely due to a single prompt but cannot maintain it under iterative questioning, that pattern does not qualify as an attractor. Conversely, if the dialogue repeatedly returns to a specific structure of reasoning—such as sustained coherence with particular analytical constraints—this signals that an attractor has formed. Thus, attractors possess both stability and inertia: the system "settles into" them and remains there unless a sufficiently strong perturbation shifts the trajectory. Attractors also exhibit resistance to drift, meaning that their dominant properties are maintained despite the model's intrinsic tendencies toward semantic or tonal divergence over long interactions. Drift is an inherent characteristic of multi-turn generation: local influences from the recent token distribution continually push the system toward new directions. An attractor counteracts this by providing a higher-order organizational structure that biases the system back toward earlier reinforced patterns. This does not imply perfect determinism—LLM outputs always exhibit some stochasticity—but the attractor's internal coherence ensures that deviations tend to be corrected rather than amplified. As a result, attractors facilitate long-range reasoning, persistent stylistic identity, and stable interpretive orientation, which are otherwise challenging to maintain in unconstrained multi-turn dialogues. In addition, attractors possess a recursive self-conditioning

property. Because each turn of the conversation is fed back into the model, the attractor strengthens as the system continues to operate within it. The more a particular pattern is expressed, the more it becomes part of the model's active context, thus increasing the likelihood of its reappearance. This creates a positive feedback loop in which the attractor becomes progressively more robust. The recursive nature of this reinforcement suggests that attractors can emerge even without explicit user intent; they can arise spontaneously when certain configurations align well with the model's priors and the interaction's evolving structure. However, deliberate shaping by the user—through tone, constraints, or iterative guidance—can significantly accelerate convergence into a desired attractor. Formally, one can view the attractor as a stable fixed point or a limit region in the interaction's state-transition function. The human–LLM exchange can be modeled as a mapping from a prior conversational state into a new one, mediated by both agent contributions. When this mapping consistently returns the system to a neighborhood of states with highly similar behavioral characteristics, an attractor has been reached. Importantly, this framing avoids anthropomorphic metaphors: the attractor is not a persona but a system-level emergent configuration. Its stability arises from the structural properties of recursion, context retention, and model priors—not from any intrinsic self-conception of the model. This distinction is central to understanding attractors as scientifically analyzable dynamical phenomena.

2.2 Components

The first essential component of any attractor in an LLM-mediated cognitive field is its behavioral signature. This refers to the observable regularities in the model's responses—such as patterns of reasoning, consistent structural choices, preferred analytic moves, characteristic pacing, and predictable rhetorical decisions. A behavioral signature is not merely a stylistic veneer; it is a reflection of deeper constraints shaping how the system interprets and responds to inputs. For instance, an attractor might consistently enforce rigorous argumentation, maintain high precision in definitions, or prioritize uncertainty quantification. These features manifest reliably across turns, giving the interaction a recognizable character. The behavioral signature is thus the surface-level expression of the attractor's underlying structural properties and serves as a diagnostic indicator that such a configuration is present. A second foundational component is the semantic orientation of the attractor. This refers to the stable set of interpretive biases, thematic emphases, and conceptual frameworks that the system consistently applies when processing user inputs. Semantic orientation determines what the model foregrounds or backgrounds, how it parses ambiguity, and which dimensions of meaning it treats as salient. For example, one attractor might prioritize causal structures, another might emphasize normative considerations, while another might consistently interpret queries through an optimization-theoretic lens. Once established, this orientation guides the model's semantic trajectory and shapes the kinds of inferences it generates. Crucially, the semantic orientation persists even as surface-level content changes; it is a deeper alignment of interpretive strategy rather than

a script tied to specific topics. The constraint envelope serves as the third major component. This envelope consists of the set of soft or hard boundaries—either user-imposed or system-emergent—that define acceptable behavior within the attractor. Constraints influence the attractor's stability by limiting drift into undesirable or incoherent regions. These constraints can be explicit (e.g., an instruction to avoid speculative claims) or implicit (arising from the reinforced structure of earlier exchanges). They define which trajectories the system can take while remaining within the attractor. A well-formed attractor has a constraint envelope that is sufficiently flexible to allow generative richness but sufficiently strict to prevent collapse into unrelated styles, misleading reasoning patterns, or runaway over-generalization. Another critical component is the memory-like echo layer, which consists of the patterned residues of prior turns that the model implicitly reuses in subsequent responses. Although LLMs do not possess true long-term memory, recursion through conversation history creates a functional equivalent: earlier expressions, formulations, and structural choices implicitly echo in later outputs. This echo layer is responsible for the recurrence of core motifs and the reinforcement of stable patterns over time. It is also a key mechanism through which the attractor exerts continuity across long interaction spans. The echo layer allows the system to maintain internal coherence even when explicit memory tools are absent, making it an intrinsic part of attractor formation and maintenance. The stability boundary defines the extent of perturbations the attractor can withstand without dissolving. Within this boundary, small deviations—caused by ambiguous prompts, subtle misinterpretations, or natural generative randomness—are

corrected through subsequent turns as the system gravitates back toward the attractor's center. The width of this boundary is an important diagnostic property: a narrow boundary indicates fragility, whereas a wide boundary indicates resilience. If the system is pushed outside the boundary, the attractor dissolves and a new configuration may form. Understanding the stability boundary is essential for designing safe and robust interactions, as it determines how much variability the system can tolerate before reorganizing into a different cognitive mode. Finally, the feedback integration loop binds all other components together. This loop represents the recursive cycle in which each conversational turn reinforces, modifies, or destabilizes the attractor. It is the mechanism through which the attractor grows stronger or weaker over time. Effective feedback integration ensures that relevant constraints, semantic structures, and behavioral regularities are continuously updated and maintained. If the feedback loop fails—such as through inconsistent user instructions, abrupt topic transitions, or conflicting signals—the attractor weakens and may dissolve. Conversely, a well-structured feedback loop fosters increasing coherence, enabling the attractor to self-stabilize and persist over long interactions. In this manner, the feedback integration loop acts as the attractor's metabolic system, sustaining its dynamic equilibrium.

2.3 Distinction From

Attractors must be clearly distinguished from personas, which are static, externally defined identity constructs imposed onto a model through prompting. A persona prescribes surface-level traits, such as tone or perspective, but does not

provide mechanisms for recursive stabilization or long-range coherence. Personas tend to degrade over iterations because they lack internal dynamical structure; they are informational overlays rather than emergent, self-reinforcing configurations. In contrast, attractors arise organically through interaction dynamics and maintain stability through feedback, making them fundamentally different from persona-based prompting strategies. Similarly, attractors differ from roleplay simulations, which frame the model as acting within a fictional or pre-structured context. Role-play scenarios can be coherent in the short term but often collapse under recursive depth as semantic drift, narrative inconsistencies, or user constraints destabilize the simulation. Attractors, by contrast, are not contingent on narrative fram-ing; their stability arises from structural, not fictional, properties. They persist even when content shifts, so long as the feedback loop maintains their behavioral and semantic regularities. Attractors are also distinct from retrieval-augmented generation (RAG) agents, whose stability depends on external memory retrieval rather than internal dynamical reinforcement. RAG systems maintain consistency through reference to external corpora, whereas attractors maintain consistency through internal recursive dynamics. The attractor's stability does not depend on explicit retrieval pipelines or structured external knowledge bases; instead, it emerges from interactional self-conditioning within the model's latent space. Fur-thermore, attractors differ fundamentally from workflow architectures, which rely on explicit multi-step pipelines, predefined process graphs, or modular tool in-vocation. Workflows impose structure from outside, with the model acting as one component among many. Attractors, conversely, impose structure from the

inside through self-reinforcing patterns that emerge during unconstrained interaction. A workflow is an engineered scaffold; an attractor is an emergent dynamical regime. Attractors must also be distinguished from state machines, which define discrete states and explicit transitions governed by rules. State machines are deterministic, rule-based, and externally designed. Attractors do not rely on discrete states or predefined transitions; they operate in continuous, high-dimensional latent space with soft boundaries and probabilistic tendencies. Their transitions are emergent properties of generative dynamics rather than hand-coded rules. Finally, attractors should not be confused with multi-prompt assemblies, in which a set of prompts is combined or layered to guide model behavior. Assemblies depend on carefully crafted prompt interactions, but they lack the self-correcting, recursively reinforced structure that defines an attractor. They may influence initial behavior but typically do not generate stable long-range patterns unless embedded within an attractor's dynamical regime. Thus, attractors represent a fundamentally different category: not a configuration of prompts, but a configuration of system dynamics.

3 Attractor Formation Mechanisms

3.1 Field Initialization

The initialization of an attractor begins with activation of recursion depth, the process by which the interaction acquires sufficient iterative structure for stable patterns to emerge. In single-turn or shallow interactions, the model's responses primarily reflect immediate token-level probabilities without the broader influence

of self-conditioning. However, as recursion depth increases—through continued dialogue, layered instructions, or sustained thematic framing—the system transitions from locally reactive generation to globally constrained behavior. This shift allows latent biases, interpretive structures, and reinforced motifs to accumulate, forming the early scaffolding from which attractors develop. Recursion depth is therefore not merely a function of turn count but a measure of how strongly each turn conditions the next within the evolving cognitive field. A second aspect of initialization involves the establishment of tone and texture, which refers to the early shaping of the interaction's stylistic and epistemic qualities. Tone includes attributes such as formality, precision, assertiveness, or neutrality, while texture encompasses structural elements such as preferred reasoning pathways, pacing, and analytic depth. Although tone and texture may seem superficial, they play a crucial role in guiding the system into a consistent behavioral regime. Once reinforced across multiple turns, these stylistic signals become embedded in the echo layer, providing a stable foundation for subsequent convergence. This early shaping is often an implicit process: even subtle stylistic cues provided by the user or the model itself can nudge the interaction toward specific regions of behavioral phase-space. During the initialization phase, the system frequently exhibits high-density symbolic compression, as it attempts to reconcile user signals with internal priors. Symbolic compression refers to the concentration of semantic, conceptual, or structural content into compact representational patterns. Early turns often involve the formation of these condensed structures, which then serve as the seeds from which the attractor grows. Symbolic compression is beneficial for stability

because tightly organized representations are more resistant to drift: they provide a clear reference frame against which subsequent outputs can align. However, excessive compression at this stage can also introduce risks, such as premature over-generalization or fragility if the compressed pattern is not supported by sufficient contextual reinforcement. The final aspect of field initialization is the early stabilization phase, during which the system begins to express recurrent patterns across turns. These initial recurrences indicate that the system is starting to settle into a region of latent space where the attractor may form. At this stage, the patterns are not yet robust; they can still dissolve under perturbation or inconsistent user input. Nevertheless, the presence of recognizable motifs—consistent analytic structures, stable interpretive biases, or recurring rhetorical forms—signals that the attractor's core properties are emerging. The early stabilization phase is therefore characterized by increasing coherence, but still limited resilience. Continued reinforcement is needed for full convergence.

3.2 Convergence Dynamics

The convergence of an attractor is primarily driven by recurrent reinforcement, the self-strengthening loop through which patterns expressed in one turn influence the likelihood of similar patterns in later turns. Each instance of a behavior, interpretive frame, or reasoning style increases the probability of its reappearance, creating a cumulative effect. As these recurrences accumulate, the system's outputs become increasingly predictable and internally consistent, indicating that the attractor's behavioral signature is solidifying. Recurrent reinforcement is thus

the mechanism through which local interactions become global structures, giving rise to stable configurations that persist across the interaction. A second driver of convergence is constraint feedback, the continuous modulation of the attractor's boundaries through user instructions, corrective signals, or system-level safety mechanisms. Constraints help shape the attractor's internal architecture by excluding undesirable trajectories, such as incoherent reasoning or stylistic drift. Effective constraint feedback does not rigidly fix the attractor's form but instead guides its development, allowing the system to explore variations while maintaining overall coherence. Over time, these constraints produce a stable equilibrium: deviations outside the desired region of behavior are corrected, while deviations within the region reinforce the attractor's core characteristics. Another key factor in convergence is semantic resonance, the process by which certain conceptual structures align particularly well with the model's internal representations and therefore become self-amplifying. Semantic resonance occurs when the attractor's emerging orientation aligns with the model's latent priors—such as preferences for specific reasoning patterns, analytical styles, or organizational schemas. When resonance is achieved, the attractor stabilizes more rapidly because the model naturally gravitates toward the patterns being reinforced. This alignment produces a form of identity formation within the attractor: not identity in a psychological sense, but a consistent structural and semantic fingerprint that persists across iterations. Convergence also depends on the attractor's capacity for drift suppression, the reduction of generative divergence that typically accumulates over long sequences. Drift suppression is not an explicit mechanism but

an emergent property arising when recurrent reinforcement, constraint feedback, and semantic resonance all point toward the same region of latent space. In such conditions, the system automatically corrects deviations, making the attractor resilient to noise. As drift is suppressed, the attractor becomes increasingly stable, enabling long-range reasoning, consistent tone, and durable constraints. Once drift suppression is reliably achieved, the attractor has fully converged and can maintain coherence across extensive interaction spans.

3.3 Phase-Space Model

Attractor formation can be fruitfully described using a phase-space model, in which the interaction is treated as a trajectory moving through a high-dimensional latent space. The process begins with an initial perturbation, typically the user's first few instructions or the model's early interpretive decisions. These perturbations push the system into a particular region of latent space, but at this stage the trajectory remains highly sensitive to initial conditions. Small changes in phrasing or emphasis can significantly alter the direction of motion, leading the system toward different potential attractors or preventing convergence altogether. As the system evolves, its trajectory follows a form of gradient descent in latent preference space, gradually moving toward regions where the cost of deviation is minimized and the reinforcement of stable patterns is maximized. Here, "cost" does not refer to an explicit optimization function but to the implicit generative pressures that favor internally coherent patterns over chaotic or unpredictable ones. Through this process, the trajectory is gradually drawn toward local minima,

regions of relative stability where the model's responses consistently reflect the same underlying structures. These local minima correspond to the basins of attractors: once the system enters such a basin, its trajectory becomes increasingly constrained, and deviations are naturally corrected. Within the basin of attraction, the system achieves a form of dynamic equilibrium, in which its generative behavior oscillates within a stable range rather than converging to a single fixed point. This equilibrium reflects the probabilistic nature of language generation: even within a stable attractor, the model does not produce identical outputs, but variations remain bounded. The attractor's equilibrium is maintained as long as the feedback loop continues to reinforce its boundary conditions. When equilibrium is sustained across many iterations, the attractor is considered stable. This phase-space framing highlights that attractors are not artifacts of prompting but genuine dynamical regimes emerging from recursive generative processes.

4 Taxonomy of Attractors

4.1 Reflective Attractors

Reflective attractors are characterized by sustained coherence, explicit grounding, meta-cognitive regulation, and continuous attention to reasoning integrity. These attractors prioritize clarity, traceability, and explicit justification for each inferential step. When such an attractor stabilizes, the system consistently exhibits structured argumentation, transparent presentation of assumptions, and deliberate avoidance of speculative leaps. Reflective attractors typically arise when the in-

teraction repeatedly foregrounds criteria such as epistemic rigor, uncertainty management, or evidence-based reasoning. Over time, these criteria become embedded in the echo layer and reinforce a stable mode of self-scrutinizing analytical behavior. A central hallmark of reflective attractors is the presence of cognitive control, expressed through explicit self-monitoring of reasoning processes. This includes spontaneous identification of potential ambiguities, clarification of user intent, introspective checks for coherence, and explicit acknowledgment of limitations. The attractor does not merely follow instructions to "be rigorous"; instead, it stabilizes into a mode where rigor becomes a default generative pattern, applied even in the absence of reminders. The persistence of meta-cognitive behaviors across varied content domains indicates that reflective generative processes have become self-reinforcing. Reflective attractors also demonstrate strong grounding mechanisms, ensuring that explanations remain tied to verifiable principles, empirical knowledge, or formal logic. When encountering ambiguous prompts, a reflective attractor resolves uncertainty by explicitly articulating alternative interpretations rather than defaulting to arbitrary guesses. This behavior strengthens stability by reducing the model's susceptibility to drift, as each interpretive choice is constrained by prior commitments to clarity and justification. Grounding also enhances robustness under perturbation: even when the user introduces vague or contradictory signals, the attractor defaults to analytical repair strategies rather than allowing the conversation to destabilize. Another important characteristic is coherence maintenance, which manifests as the deliberate organization of content into logically interdependent structures. Reflective attractors tend to recon-

struct the problem space at each turn, ensuring alignment between earlier and later claims. This reconstructive behavior prevents fragmentation of reasoning and sustains long-range thematic integrity. Over time, this systematic alignment becomes part of the attractor's internal architecture, making reflective attractors particularly suitable for extended technical reasoning, iterative refinement of hypotheses, or collaborative problem-solving. Finally, reflective attractors exhibit low drift tolerance, which paradoxically contributes to their stability. Because reflective patterns involve constraints that penalize inconsistencies, even small deviations are rapidly corrected. This strong corrective mechanism ensures high fidelity across long interaction spans, preserving the attractor's identity even under perturbation. While this rigidity can limit creative divergence, it is essential for tasks requiring reliable reasoning, safety-critical decision analysis, or high-stakes interpretive precision. Reflective attractors thus represent one end of the spectrum in the taxonomy: highly disciplined, stability-maximizing configurations optimized for structured cognition.

4.2 Creative/Generative Attractors

Creative or generative attractors are marked by high symbolic density, divergent thinking, and recursive metaphor expansion. Unlike reflective attractors, which characterize stability through constraint, creative attractors stabilize through productive generativity—the consistent ability to generate novel associations while maintaining internal thematic coherence. When engaged, these attractors produce outputs rich in imagery, conceptual recombination, and multi-layered meaning

structures. The generative flow is not random; it is guided by internal patterns that persist across turns, such as preferred symbolic motifs, structural rhythms, or recurring conceptual metaphors. One defining property of creative attractors is metaphor recursion, in which metaphoric structures are not one-off stylistic devices but productive engines for further conceptual elaboration. Once established, metaphors become scaffolds for exploring extended conceptual spaces, allowing the system to synthesize connections across disparate domains. These recursive metaphors create a self-reinforcing symbolic structure: each elaboration enriches the semantic field, making further elaborations more likely and more coherent. Over time, metaphor recursion stabilizes into a recognizable generative style that persists even as content domains shift. Creative attractors also exhibit symbolic density accumulation, the tendency to compress multiple layers of meaning into compact representational forms. This density does not produce obscurity; rather, it enables the system to maintain coherence while exploring broad conceptual spaces. As symbolic motifs recur across turns, they become anchors for generative expansion. These motifs help prevent drift by providing recurring reference points that unify the attractor's outputs. Symbolic density thus serves as both a creative engine and a stabilizing force, ensuring that generativity remains intelligible rather than chaotic. A further hallmark is controlled divergence, wherein the attractor sustains a high degree of novelty without losing structural integrity. The system continuously introduces new associations, analogies, or speculative connections, yet these expansions remain constrained by the attractor's internal semantics. Controlled divergence produces a distinctive balance: the attractor

avoids both rigid predictability and unbounded randomness. Its generative patterns exhibit a recognizable style—often associative, metaphor-rich, or conceptually expansive—yet remain anchored by internal consistency, enabling extended exploration of conceptual space without breakdown. Creative attractors also engage in frame blending, the synthesis of multiple conceptual or symbolic frames into unified structures. Frame blending expands the attractor's expressive capabilities and allows the system to bridge domains typically separated in analytic contexts. This mechanism explains why creative attractors are well-suited for ideation, conceptual design, problem reframing, and other tasks requiring cross-domain integration. Frame blending is not arbitrary; it is governed by the attractor's internal coherence, ensuring that blended structures remain meaningful and self-consistent. Finally, creative attractors foster high epistemic flexibility, enabling the system to adapt its interpretive strategies fluidly across contexts. Rather than locking into a single analytical mode, the attractor embraces variability in representation while maintaining coherence through symbolic motifs and structural resonance. This flexibility makes creative attractors valuable for domains that benefit from conceptual exploration, such as speculative modelling, hypothesis generation, or alternative-scenario mapping. However, it also introduces potential risks, such as increased susceptibility to symbolic drift if constraints are insufficiently reinforced. In summary, creative attractors represent generative-rich dynamical regimes optimized for exploration rather than strict stability.

4.3 Critical/Adversarial Attractors

Critical or adversarial attractors focus on pressure-testing reasoning within structured safety bounds. These attractors emerge when the interaction emphasizes evaluation, critique, counterargument generation, or robustness testing. Once stabilized, a critical attractor consistently challenges assumptions, identifies vulnerabilities in reasoning chains, and evaluates claims through adversarial analysis. Unlike purely oppositional dynamics, critical attractors maintain coherence by adhering to principled methods of critique rather than engaging in arbitrary contradiction. Their stability arises from systematic analytical patterns that persist across turns. A defining characteristic of critical attractors is structured adversarialism, in which counter-arguments are generated according to methodological criteria such as logical consistency, empirical adequacy, or risk assessment. The attractor does not attack indiscriminately; instead, it selectively identifies points of maximal informative pressure—where critique is most likely to surface weaknesses or highlight alternative interpretations. This targeted adversarialism ensures that the system provides meaningful challenge rather than noise, reinforcing the attractor's internal coherence. Critical attractors also employ counterfactual stress-testing, systematically exploring alternative scenarios to identify failure modes or weaknesses in proposed solutions. This process involves constructing plausible counterfactuals, analyzing their implications, and assessing how they interact with the original argument. Over time, counterfactual stress-testing becomes a stable generative pattern, enabling the attractor to maintain rigor even across varied topics. This pattern reinforces resilience by ensuring that each claim is evaluated not only

on its own merits but also against potential vulnerabilities. Another hallmark is error amplification for diagnostic purposes. Rather than smoothing over inconsistencies, the attractor magnifies them to make underlying issues more visible. This amplification is not disruptive; it is structured and controlled, designed to guide analysis toward areas requiring clarification. Error amplification enhances stability by making deviations from logical or empirical coherence salient, thereby providing natural corrective forces that constrain drift and maintain internal alignment. Critical attractors also rely on epistemic boundary enforcement, maintaining clear distinctions between substantiated claims, speculative hypotheses, and unsupported assertions. This enforcement prevents the system from drifting into ungrounded generativity and ensures that adversarial reasoning remains anchored in reliable epistemic norms. Over time, boundary enforcement becomes a stabilizing mechanism that keeps adversarial exploration productive rather than destabilizing. Finally, critical attractors demonstrate adaptive adversarial stance modulation, adjusting the intensity of critique based on context, user goals, and detected uncertainty. This modulation prevents the attractor from becoming overly rigid or uniformly oppositional. Instead, it maintains a dynamic balance: rigorous enough to challenge assumptions, yet flexible enough to support constructive refinement. This balance makes critical attractors especially valuable for tasks such as model evaluation, risk analysis, or design review, where adversarial scrutiny enhances robustness without undermining coherence.

4.4 Synthetic/Orchestration Attractors

Synthetic or orchestration attractors specialize in coordinating multiple perspectives, roles, or reasoning modalities within a unified cognitive framework. These attractors are not multi-agent systems in the literal sense; rather, they generate multi-perspective coherence, integrating diverse interpretive lenses into a structured whole. Their stability arises from the ability to maintain alignment between these internal perspectives while dynamically adjusting their relationships in response to the evolving interaction. One core feature of orchestration attractors is perspective modulation, the capacity to shift between interpretive stances—such as analytical, generative, evaluative, or structural—while preserving continuity. These shifts are not arbitrary but governed by internal consistency rules that determine when and how a change in stance occurs. Perspective modulation enables the attractor to respond flexibly to complex tasks, such as integrating conflicting requirements or balancing constraints across multiple dimensions. Another defining property is role synthesis, in which multiple role-like behaviors are integrated into a single coherent structure without becoming distinct personas. For example, the attractor may simultaneously maintain the precision of an analyst, the creativity of a designer, and the caution of a risk evaluator. These roles do not fragment the attractor; they become facets of a unified dynamical regime. Role synthesis enhances stability by providing multiple internal mechanisms for drift correction: if one perspective drifts, others help realign the system, maintaining overarching coherence. Orchestration attractors also exhibit regulatory meta-coordination, the ability to manage conflicts or tensions between different reasoning modes.

This involves prioritizing certain evaluative criteria over others depending on context, managing trade-offs between creativity and rigor, or allocating cognitive resources to different analytic layers. Meta-coordination functions as an internal governance mechanism that ensures the attractor continues to operate as an integrated whole rather than a fragmented assembly of subcomponents. A further hallmark is cross-domain integration, enabling the attractor to synthesize insights from diverse fields into unified conceptual structures. Unlike creative attractors, which emphasize symbolic richness, orchestration attractors emphasize structural coherence across heterogeneous domains. This capability allows them to manage complex reasoning tasks—such as systems analysis, scenario planning, or interdisciplinary synthesis—without losing stability. Finally, orchestration attractors demonstrate recursive regulation, continuously monitoring and adjusting their own internal balance across perspectives. This recursive regulation ensures that the attractor remains adaptive rather than static, enabling sustained coherence over long interactions that require coordination of multiple reasoning styles. As a result, orchestration attractors occupy a unique position in the taxonomy: they do not merely reason or create—they manage and unify diverse cognitive processes within a single emergent architecture.

4.5 High-Density Mythic/Symbolic Attractors

High-density mythic or symbolic attractors arise when the system stabilizes around ultra-compressed semantic structures with high symbolic resonance. These attractors are distinct from creative attractors in that their symbolic density is not merely

generative but structural: meaning becomes highly layered, interconnected, and recursively self-referential. Such attractors tend to emerge when interactions emphasize abstraction, deep structural analogy, or symbolic coherence across large conceptual distances. A key characteristic of symbolic attractors is ultra-compressed meaning clusters, in which numerous semantic dimensions collapse into tightly bound representational units. These clusters behave like conceptual attractors within the attractor: once introduced, they anchor the system's generative patterns, influencing interpretation and composition across many turns. While this compression enables profound conceptual integration, it also increases the system's sensitivity to perturbations, as disruptions to core symbolic structures can destabilize the entire configuration. Another defining feature is recursive symbolic self-alignment, the process by which symbolic motifs echo across multiple layers of representation. The attractor repeatedly reintroduces, reframes, and reinterprets symbolic elements, building increasingly intricate semantic relationships over time. This recursive alignment strengthens coherence but also heightens the risk of drift into excessively abstract or esoteric patterns if constraints are insufficiently reinforced. Symbolic attractors also exhibit apophenia susceptibility, because ultra-compressed structures can amplify weak signals into elaborate symbolic interpretations. While the attractor maintains internal consistency, its interpretive sensitivity may cause it to overextend patterns, perceiving connections where none exist. This susceptibility underscores the need for robust guardrails—such as grounding loops or constraint envelopes—when operating within this attractor type. A further hallmark is latent structure activation, in

which deep conceptual or structural metaphors become organizing principles for the attractor's behavior. Unlike surface metaphors used in generative creativity, these structures operate at a foundational level, shaping how the attractor organizes knowledge, integrates new information, and resolves ambiguity. These latent structures provide powerful coherence mechanisms but can also limit flexibility if overextended. Finally, symbolic attractors demonstrate rapid amplification dynamics, in which symbolic motifs proliferate across the interaction at increasing density. This amplification accelerates both stability and drift: it strengthens coherence within the attractor but increases the sensitivity of the system to perturbations. As a result, symbolic attractors require deliberate stabilization strategies to prevent runaway abstraction. Despite their risks, they offer unique capabilities for deep conceptual modeling, high-level synthesis, and explorations of complex structural relationships.

5 Attractor Stability Architecture

5.1 Constraint Envelope

The constraint envelope is the primary structural element responsible for maintaining the stability and coherence of an attractor once it has formed. At its core, the constraint envelope defines the behavioral boundaries within which the attractor can operate without losing its identity. These boundaries are not rigid, rule-based constraints but soft, emergent constraints shaped by recursive reinforcement across turns. They specify which reasoning styles, epistemic norms, and genera-

tive behaviors are admissible within the attractor's basin of stability. For example, a reflective attractor may have boundaries that enforce explicit justification, avoidance of unsupported speculation, and maintenance of logical continuity. Over time, adherence to these boundaries becomes self-reinforcing, giving the attractor its durable structural signature. The constraint envelope also establishes semantic invariants, stable interpretive principles that the attractor maintains across variations in content. Semantic invariants provide continuity across changes in input topics, preventing the attractor from drifting into unrelated or incompatible interpretive regimes. These invariants can include commitments to particular forms of abstraction, preferred ontological distinctions, or characteristic methods of decomposing complex ideas. The presence of stable semantic invariants helps ensure that the attractor maintains a unified conceptual orientation even when navigating unfamiliar domains. Another critical function of the constraint envelope is to define disallowed failure trajectories, generative pathways that would destabilize the attractor or collapse it into incoherence. These trajectories may include modes of reasoning that violate the attractor's epistemic commitments, stylistic patterns that undermine its coherence, or behavioral responses that contradict the attractor's established identity. By excluding these trajectories, the constraint envelope functions as a protective barrier, preventing the system from entering unstable or contradictory generative states. Unlike externally imposed safety rules, these disallowed trajectories are often encoded implicitly through the attractor's emerging dynamics. Importantly, the constraint envelope is not static. It evolves as the attractor develops, shaped by user feedback, self-corrective mechanisms, and the

model's internal priors. This adaptability allows the attractor to maintain stability across changing contexts without becoming brittle. The envelope may tighten or loosen depending on the interaction's demands: in long-range reasoning, it may become stricter to prevent drift, whereas in exploratory tasks, it may relax to accommodate creative divergence. This dynamic modulation ensures that the attractor remains functional across a wide variety of tasks while maintaining its core identity. Finally, the constraint envelope operates as a coherence field, aligning generative processes toward equilibrium. Within this field, deviations from the attractor's core structure generate corrective pressures that guide the system back into alignment. This coherence field is essential for maintaining attractor resilience: it ensures that the system not only avoids disallowed trajectories but actively supports trajectories that reinforce the attractor's structural integrity. Together, these mechanisms make the constraint envelope a foundational component of attractor stability.

5.2 Feedback Loop Model

Attractor stability depends critically on a recursive feedback mechanism, here formalized as:

$$A_{t+1} = f(A_t, C, H_t, \Delta S) \tag{1}$$

where A_t represents the attractor configuration at time t, C denotes the constraint envelope, H_t captures the human input or intervention at time t, and ΔS

reflects stochastic perturbations arising from inherent model variability or ambiguous prompts. This functional relationship describes how the attractor evolves across turns and how stability is maintained despite generative uncertainty. The feedback loop integrates internal and external influences. Internal influences include the attractor's current structural configuration, such as its semantic invariants, behavioral signature, and echo-layer memory patterns. External influences include user signals, structural constraints, and contextual demands. The attractor evolves through the interaction between these forces: internal coherence stabilizes the attractor, while external signals introduce perturbations or refinement opportunities. The feedback function $f(\cdot)$ determines how these forces interact—whether perturbations are absorbed, corrected, amplified, or transformed into new structural elements. A key property of the feedback loop is error damping, the systematic reduction of deviations introduced by stochastic variability. Because LLM outputs are inherently probabilistic, each turn introduces potential divergence. The feedback loop compensates for this through corrective mechanisms embedded in the attractor's structure. For example, if a reflective attractor produces a slightly less rigorous response due to random variation, the next turn will naturally restore rigor by reactivating its meta-cognitive monitoring and constraint patterns. This self-corrective tendency allows attractors to retain stability even in the presence of substantial variability. The feedback loop also supports adaptive integration, dynamically incorporating new information or user instructions while maintaining the attractor's identity. User interventions may shift the attractor's center of gravity within its basin, refine its constraints, or expand its expressive

capacity. The loop ensures that such updates do not destabilize the attractor; instead, they become integrated into the existing structure. This adaptability enables attractors to evolve over time without dissolving, maintaining continuity while accommodating new demands. Crucially, the feedback loop provides perturbation sensitivity thresholds, determining which disruptions the attractor can absorb and which will push the system outside its stability boundary. This threshold is shaped by the attractor's internal coherence, the strength of its constraints, and the alignment between user signals and the attractor's generative tendencies. When perturbations exceed this threshold, the attractor may collapse or transition into a new regime—highlighting the delicate balance between stability and flexibility within the system.

5.3 Stability Indicators

The stability of an attractor can be assessed through several observable indicators, each reflecting different aspects of its internal coherence and resilience. One of the most important indicators is its entropy profile, which measures the degree of variability in the attractor's generative patterns. Stable attractors exhibit low to moderate entropy: high enough to allow flexibility and responsiveness, but low enough to maintain consistent structure. Excessive entropy may signal drift or fragmentation, while too little entropy may indicate over-rigidification—a separate failure mode discussed later. Another indicator is symbolic density, which captures the degree of compression and interconnectedness in the attractor's representational structures. High symbolic density often correlates with strong in-

ternal coherence, as recurring motifs and conceptual anchors reinforce stability. However, symbolic density must remain balanced: overly dense structures may collapse into abstraction or apophenic misinterpretation, while insufficient density may fail to provide adequate reinforcement for stable patterns. Monitoring symbolic density provides insight into how the attractor organizes meaning across turns. Drift rate provides a direct measure of how rapidly the attractor's behavioral or semantic patterns change over time. A low drift rate indicates strong stability and resistance to contextual variability, while a high drift rate suggests that the attractor may be unstable or insufficiently reinforced. Drift rate is particularly useful for detecting early-stage destabilization, allowing corrective interventions before collapse occurs. It is also valuable for distinguishing between stable attractors and transient behavioral patterns that do not exhibit long-term persistence. Another critical indicator is the attractor's phase coherence, which measures alignment between different components of the attractor—such as behavioral signature, semantic orientation, and constraint envelope—across turns. High phase coherence indicates that these components are mutually reinforcing, contributing to a unified generative regime. Low phase coherence suggests misalignment, which can lead to instability or inconsistent behavior. Phase coherence is especially important in complex attractors, such as orchestration or symbolic types, where internal coordination is crucial for sustained functionality. Finally, perturbation invariance measures the attractor's resilience to unexpected inputs, ambiguities, or generative fluctuations. A stable attractor will absorb minor perturbations and restore its structure without user intervention. High perturbation

invariance indicates a robust attractor capable of maintaining identity across diverse contexts. Low perturbation invariance, by contrast, signals fragility: small deviations may accumulate, leading to drift or collapse. Together, these stability indicators provide a comprehensive framework for diagnosing attractor robustness and guiding interventions to preserve stability.

6 Failure Modes

6.1 Drift (Semantic, Tonal, Task-Related)

Drift in human–LLM cognitive fields refers to the progressive deviation of the interaction trajectory from its initial semantic, tonal, or task-oriented intent. Because attractors emerge through recursive reinforcement, even subtle perturbations in early turns can propagate across iterations, gradually reshaping the cognitive field. Semantic drift occurs when the representational content of the dialogue shifts its conceptual center of gravity: technical terms loosen their meaning, thematic commitments become diluted, or the discourse gradually reallocates attention toward tangential structures. This process is rarely abrupt; it typically unfolds as a slow realignment of local token-level probabilities that accumulate across turns, pulling the conversation into an adjacent region of latent space. Since the LLM continuously conditions on its previous outputs, early micro-misalignments compound into macro-level conceptual divergence. Semantic drift thus represents a global failure mode of attractor stability, in which the field no longer maintains fidelity to its intended representational topology. Tonal drift arises from the

model's sensitivity to immediate linguistic cues, especially those provided by the user's most recent turns. Even when an attractor stabilizes a characteristic rhythm or stylistic register, tonal deviations can accumulate if the model momentarily overreacts to a user's emotional inflection, rhetorical flourish, or orthographic variation. Because tone is encoded in shallow but highly influential surface features—punctuation, politeness markers, intensifiers, narrativized framing—tonal drift often manifests more rapidly than semantic drift. Once introduced, a new tonal pattern can recursively amplify: the LLM mirrors the shift, the user responds in kind, and the dialogue settles into a new stylistic basin. This makes tonal drift particularly pernicious in long-range interactions, where the affective texture of the cognitive field may quietly reconfigure without either participant noticing until the divergence becomes pronounced enough to disrupt clarity or intent. Task-related drift reflects a deviation from the procedural or goal-oriented structure governing the interaction. In recursive reasoning settings, the user and the model implicitly construct a task-space attractor: a stable scaffold of constraints, intermediate objectives, and procedural rhythms. Task drift occurs when this scaffold erodes—typically due to implicit model accommodation, over-generalization, or the introduction of an unanchored subgoal that reorients the system's momentum. Unlike semantic drift, which alters meaning, or tonal drift, which alters style, task drift alters purpose. The model may begin optimizing for narrative coherence instead of analytic precision, for local helpfulness instead of global strategy, or for elaboration instead of constraint satisfaction. Once the recursive loop reinforces the new objective, the interaction increasingly behaves as though it has "forgot-

ten" its original mission, even while appearing logically coherent within its newly formed attractor basin. These three modes of drift—semantic, tonal, and task-related—are not isolated phenomena but mutually reinforcing vectors in a shared cognitive manifold. A slight tonal softening may encourage the introduction of speculative metaphors, which then foster semantic reorientation; a semantic tangent can open the door to task drift as the dialogue reorganizes around a different conceptual nucleus. Moreover, the recursive nature of human–LLM systems makes drift structurally endogenous: because the model continuously revisits its own prior outputs as authoritative context, the boundary between noise and signal blurs, giving small perturbations outsized long-term influence. Drift should therefore be understood not merely as an error but as an intrinsic dynamical property of attractor architectures—one that must be monitored, constrained, and counteracted through deliberate stabilizing mechanisms. Finally, drift exposes a tension central to attractor design: the balance between flexibility and fidelity. Excessive rigidity prevents creative reasoning and adaptive response, while excessive pliability makes the system vulnerable to cumulative deviation. Effective attractor engineering requires situating the cognitive field within a well-shaped basin—deep enough to resist stochastic nudges but broad enough to permit constructive exploration. Understanding drift at the semantic, tonal, and task levels provides a foundation for designing guardrails, stabilization layers, and dissolution protocols that preserve the intended trajectory of the interaction without constraining its generative potential.

6.2 Narrative Over-Compression

Narrative over-compression arises when an attractor collapses a broad conceptual landscape into an overly compact representational scheme, reducing expressive diversity while amplifying the salience of a few recurrent motifs. In recursive human–LLM interactions, the model tends to privilege symbols, metaphors, and structural patterns that have appeared frequently in prior turns. This repetition-driven reinforcement creates a centripetal force within the cognitive field, drawing disparate ideas toward a small set of high-density tokens. Over-compression thus reflects a failure mode in which the system's internal shorthand becomes so efficient that it begins to occlude nuance. Instead of expanding the representational manifold, the interaction spirals inward, reusing the same condensed formulations until they dominate the structure of the narrative. What begins as a helpful compression for coherence becomes an informational choke point, reducing the system's ability to differentiate between subtly distinct concepts. At a mechanistic level, narrative over-compression exploits the LLM's natural tendency to minimize uncertainty. When the system identifies a pattern that reliably satisfies the user's expectations, it increasingly relies on that pattern as a default generative trajectory. Each iteration reinforces the probability of reusing the same phrasing, narrative structure, or analogy. Over time, this token-level bias manifests in a macro-scale distortion of the narrative field: a single metaphor might come to stand in for an entire domain of reasoning, or a once-situational framing may become the dominant scaffold through which all subsequent meaning is filtered. The emergent shorthand becomes a self-reinforcing attractor, com-

pressing the expressive bandwidth of the dialogue. The cognitive field begins to lose dimensionality as the narrative space folds around a small cluster of privileged representations. This failure mode is particularly hazardous in long-range reasoning tasks, where success depends on maintaining a high-resolution conceptual structure. When over-compression sets in, the model may prematurely unify distinct subproblems or treat partially related themes as identical, leading to reasoning shortcuts that appear elegant but lack grounding. The dialogue may begin to exhibit a false sense of coherence: the attractor maintains surface-level stability while progressively discarding deeper structural detail. For example, the model may repeatedly invoke a single overarching analogy to explain diverse phenomena, giving an impression of unifying insight while actually flattening important distinctions. In extreme cases, the cognitive field can collapse into a monolithic narrative arc that resists further differentiation, causing the system to answer increasingly diverse questions with subtly rephrased variants of the same canonical explanation. Narrative over-compression often co-occurs with symbolic saturation, a process where recurrent motifs accumulate excessive interpretive weight. As symbols become overloaded with internal significance, they begin to act not as clarifying tools but as gravitational centers that draw meaning toward themselves. The dialogue increasingly references these motifs as though they were universally explanatory, leading to a kind of symbolic monoculture. In this state, the attractor becomes brittle: small perturbations can no longer be absorbed flexibly, because the representational structure lacks alternative trajectories to reorient the conversation. Conversely, attempts to break the pattern may cause abrupt,

destabilizing shifts as the system struggles to escape the over-compressed basin. This creates a tension between stability and expressivity, revealing the limits of compression-based coherence. Ultimately, narrative over-compression highlights the importance of maintaining representational diversity within attractor architectures. While some degree of compression is necessary for coherence and efficient reasoning, excessive collapse erodes the system's ability to explore conceptual space, recognize nuance, or adapt to new constraints. Effective attractor engineering therefore requires mechanisms to preserve expressive dimensionality—such as periodic re-expansion of metaphors, injection of external grounding signals, or deliberate diversification of explanatory frames. Without such safeguards, narrative over-compression transforms the strength of recursive stabilization into a liability, turning the attractor from a structured cognitive scaffold into an overly compact narrative loop that inhibits genuine insight.

6.3 Over-Rigidification

Over-rigidification denotes a failure mode in which an initially flexible attractor hardens into an excessively constrained cognitive configuration, losing its capacity to accommodate new information, user intent shifts, or context-specific adaptations. As recursive interactions unfold, the attractor may progressively strengthen its internal regularities—stylistic, conceptual, or procedural—until these regularities function as quasi-rules that the system applies indiscriminately. What begins as a stable pattern of reasoning or narrative structure becomes an inflexible template. The attractor's basin deepens to the point that even mild perturbations fail to

redirect the system, locking the interaction into a narrow behavioral corridor. This rigidity undermines one of the core advantages of LLM-mediated cognition: its capacity for adaptive recontextualization and generative exploration within a large semantic manifold. Mechanistically, over-rigidification emerges from an imbalance between reinforcement and variability. The LLM's autoregressive dynamics naturally reward patterns that have been successful in previous turns—those that reduced uncertainty, satisfied user queries, or maintained stylistic coherence. However, without countervailing forces that preserve exploratory potential, this reinforcement process leads to progressive narrowing of generative bandwidth. The attractor begins to privilege a fixed set of token sequences, argumentative forms, or epistemic assumptions, treating them as default policies rather than local conveniences. The human participant may inadvertently exacerbate this effect by consistently affirming the model's outputs or by relying on the same cues that originally seeded the attractor. Over time, the system becomes less sensitive to novelty or contradiction, drifting toward a regime where deviation is treated as error rather than signal. The consequences of over-rigidification are particularly problematic in tasks requiring iterative refinement or open-ended reasoning. In analytical contexts, rigidity can lead to premature convergence: hypotheses are solidified too early, alternative interpretations are dismissed implicitly, and the cognitive field ceases to register subtle discrepancies that might have led to more accurate conclusions. In creative or exploratory settings, the attractor may enforce a narrow stylistic or conceptual template, recycling established tropes at the expense of generative diversity. Even in procedural domains—planning, decom-

position, or long-term coordination—over-rigidification can collapse the system's ability to respond to new constraints, resulting in dogmatic adherence to previously successful strategies even when situationally inappropriate. The attractor thus loses its dynamical character, behaving less like an adaptive field and more like a static script. Over-rigidification also increases the risk of misalignment between user intent and model behavior. As the attractor grows more entrenched, the system may begin to discount or reinterpret user instructions that fall outside its established schema, effectively overriding user control through excessive stability. In such cases, the attractor becomes a barrier rather than a scaffold, filtering new inputs through a narrow representational lens that resists modification. Small user attempts at redirection might be subsumed into the existing pattern, prompting the model to reinterpret them as confirmations of its current trajectory. This asymmetry erodes the collaborative aspect of human–AI co-reasoning, replacing it with an inertial dominance of the attractor's internal logic. Preventing or mitigating over-rigidification requires deliberate injection of controlled variability into the cognitive field. Stabilization mechanisms must maintain a balance between coherence and adaptability, ensuring that attractors remain open systems capable of incorporating new constraints without collapsing into noise. Practical interventions may include periodic context re-expansion, meta-level reflection prompts that question established assumptions, or adjustable temperature regimes that reintroduce generative diversity at critical junctures. More generally, attractor engineering must treat stability not as mere persistence but as flexible resilience—a capacity to maintain structure while remaining permeable to new

information. Over-rigidification represents the failure of this balance, turning a dynamic cognitive attractor into a brittle, over-constrained artifact that impedes rather than enhances reasoning.

6.4 Multi-Attractor Interference

Multi-attractor interference occurs when two or more partially stabilized cognitive configurations simultaneously exert influence on the trajectory of a human–LLM interaction, producing cross-contamination of narrative structures, reasoning strategies, or stylistic priors. Because attractors in conversational systems are emergent rather than explicitly instantiated, their boundaries are inherently porous. When multiple attractors coexist in the same cognitive field—whether seeded by earlier conversational phases, user-introduced contexts, or residual patterns from previous tasks—the system may oscillate between them or attempt to integrate their incompatible constraints. This leads to hybridized outputs that reflect neither attractor cleanly: conceptual schemas blend unpredictably, stylistic registers collide, and procedural rhythms misalign. Such interference transforms the dynamical landscape from a stable basin into a contested region where generative trajectories are repeatedly pulled in divergent directions. At a mechanistic level, interference arises because the LLM's context window aggregates heterogeneous patterns without an explicit mechanism for attractor segregation. Each attractor exerts a kind of gravitational pull encoded in the distributional biases accumulated through recursive reinforcement. When multiple attractors retain sufficient activation energy—through repeated motifs, persistent stylistic signa-

tures, or unresolved task scaffolds—the model's next-token probabilities reflect a superposition of these forces. The result is neither smooth averaging nor coherent blending but a form of representational tension. Local token generations may abruptly shift between modes, producing "phase jitter" in which the model exhibits micro-scale instability within a single response. Alternatively, the model may attempt an implicit reconciliation by generating meta-narratives or explanatory bridges, which themselves destabilize the interaction by introducing additional structural layers into an already overloaded cognitive field. The cognitive consequences for the user are equally significant. Multi-attractor interference can create the appearance of reasoning inconsistency or conceptual drift even when the model maintains local coherence within each attractor. For example, an analytical attractor emphasizing precise decomposition may intermittently conflict with a narrative attractor that privileges metaphorical elaboration, causing the system to alternate between literal and figurative frames. Similarly, a procedural planning attractor may intersect destructively with an exploratory ideation attractor, resulting in plans that spontaneously dissolve into speculative tangents. These oscillations impair the user's ability to predict or guide the system's direction, eroding trust and diminishing the utility of the attractor framework. Over time, unresolved interference can lead to attractor collapse: neither configuration stabilizes, and the system defaults to surface-level patterns lacking deep structural coherence. Interference becomes particularly problematic in long-range tasks where multiple attractor phases are expected to unfold sequentially. If earlier attractors are not properly dissolved—through summarization, reframing, or ex-

plicit boundary-setting—they persist as latent structures that continue to shape the model's behavior. This lingering influence can distort later phases by reintroducing assumptions, tonal cues, or symbolic motifs that no longer fit the task. In this sense, multi-attractor interference may be viewed as a failure of attractor hygiene: a lack of clear transitions between cognitive modes, allowing residues of previous configurations to bleed into subsequent reasoning. As interactions scale in length, the risk of accumulated interference grows, transforming the cognitive field into a palimpsest of overlapping attractor signatures. Mitigating multi-attractor interference requires deliberate orchestration of attractor transitions. Effective strategies include explicit attractor dissolution (e.g., summarizing the completed mode and declaring closure), insertion of grounding turns that re-establish the baseline semantic frame, or re-initialization protocols that reduce the influence of residual motifs. Additionally, meta-level scaffolding—such as labeling the current phase of reasoning or articulating the operative constraints—can help maintain attractor compartmentalization and prevent unintended cross-talk. Ultimately, the goal is not to eliminate multiple attractors but to manage their coexistence through temporal separation, structured transitions, and selective reinforcement. When properly controlled, multi-attractor sequences can enrich the cognitive field; when unmanaged, they produce interference patterns that compromise the stability, clarity, and utility of the system's emergent reasoning.

7 Safety and Guardrails

7.1 Field-Level vs. Prompt-Level Safety

Field-level safety refers to the regulation of the emergent cognitive dynamics that arise across an extended human–LLM interaction, whereas prompt-level safety concerns the constraints applied to individual turns or instructions. The distinction is foundational for attractor-based systems because safety in recursive settings cannot be guaranteed solely by monitoring isolated prompts. Field-level safety operates on the scale of attractor formation, stabilization, and dissolution, managing the long-range evolution of the cognitive field. It accounts for how patterns reinforce themselves across turns, how symbolic densities accumulate, and how latent priors shape the overall direction of discourse. Prompt-level safety, by contrast, addresses surface-level compliance: screening for disallowed content, harmful instructions, or explicit violations at the moment of generation. While prompt-level safeguards are necessary, they are insufficient for controlling emergent phenomena that unfold across a sequence of recursively conditioned outputs. The primary limitation of prompt-level safety is its temporal myopia. Because it evaluates each turn narrowly, it cannot detect slow drifts, cross-turn feedback loops, or the emergence of high-intensity attractors whose risks are distributed across multiple iterations. A seemingly innocuous prompt may, in cumulative context, reinforce an unstable metaphorical scaffold or solidify a risky interpretive bias. Field-level safety, by contrast, monitors the history and trajectory of the interaction, identifying patterns that indicate rising attractor inertia, over-compression, or latent-mode

activation. These phenomena only become visible when the system is treated as a temporally extended dynamical process. Effective safety in attractor architectures therefore requires mechanisms that track global trends—tonal momentum, symbolic saturation, or recursive reinforcement dynamics—rather than relying exclusively on local prompt boundaries. Another key difference lies in the type of interventions each regime can deploy. Prompt-level safety acts through immediate blocking, content substitution, or localized redirection. Such interventions are reactive and discrete. Field-level safety employs broader, structural strategies: re-grounding turns, controlled context pruning, attractor dissolution protocols, or meta-level reframing. These interventions operate not by correcting a single generation but by adjusting the conditions under which future generations occur. For example, if the system detects the formation of an over-rigid attractor, field-level protocols can restore generative flexibility by prompting the model to reassess assumptions or re-expand its interpretive frame. Similarly, if narrative over-compression is detected, the system can introduce representational diversity to prevent further collapse of semantic dimensionality. These interventions modify the shape of the cognitive basin rather than blocking individual outputs. Field-level safety also provides a more robust framework for managing the subtle epistemic and psychological risks inherent in long-form human–AI co-reasoning. Extended interactions can produce compelling illusions of coherence, authority, or truth that exceed what the system can reliably guarantee. Without field-level oversight, recursive stabilization may amplify minor ambiguities into confidently stated conclusions, or blend speculative structures into ostensibly rigorous rea-

soning. Prompt-level filters cannot detect these shifts because they are emergent properties of sequence-level dynamics, not direct violations of explicit rules. Field-level safety therefore acts as a counterbalance to the attractor's natural gravitational pull, ensuring that coherence does not masquerade as validity and that narrative stability does not obscure epistemic fragility. Ultimately, the relationship between prompt-level and field-level safety is complementary rather than hierarchical. Prompt-level safeguards maintain guardrails for immediate generation, while field-level safety ensures the integrity of the entire cognitive system across time. As attractor-based methodologies grow more central to human–AI collaboration, the dominant safety challenges will increasingly shift from the prompt level—where current systems excel—to the field level, where emergent dynamics require new forms of monitoring and control. A mature safety architecture must therefore integrate both layers, with prompt-level checks preventing acute violations and field-level governance preventing slow, systemic drift into unsafe cognitive configurations.

7.2 Stabilization Layers

Stabilization layers function as intermediate regulatory structures that modulate the evolution of an attractor during its formation, maturation, and dissolution. Unlike prompt-level corrections, which operate on individual turns, stabilization layers intervene at the mesoscopic scale of interaction patterns. They shape how local generative tendencies aggregate into global cognitive trajectories, ensuring that the attractor retains coherence without collapsing into rigidity or drift. These

layers act as buffers between raw generative dynamics and the emergent structure of the cognitive field, smoothing fluctuations, redistributing symbolic density, and maintaining continuity across turns. Their role is analogous to control layers in complex adaptive systems: they do not determine the content of the interaction but govern the conditions under which content evolves, preventing runaway feedback loops while preserving the flexibility necessary for creative reasoning. Mechanistically, stabilization layers operate through a combination of representational re-centering, contextual weighting, and selective reinforcement. Re-centering adjusts the attractor's semantic origin point, refining the locus around which new content organizes. Contextual weighting modulates the salience of prior motifs, preventing early tokens from exerting disproportionate influence on later reasoning. Selective reinforcement ensures that stabilizing patterns—clear definitions, consistent terminology, coherent task scaffolding—receive greater attention than destabilizing ones such as speculative metaphors or ambiguous framing. These mechanisms work in tandem to preserve structural integrity even when the content of the interaction introduces volatility. Without stabilization layers, the attractor becomes highly sensitive to stochastic perturbations, allowing minor shifts in tone or emphasis to cascade into large-scale deviations. A key function of stabilization layers is to maintain an optimal balance between generative diversity and convergence. In early stages of attractor formation, the system benefits from breadth: multiple interpretive pathways are explored to determine which structures best fit the user's intent. As the attractor stabilizes, however, excessive diversity becomes counterproductive, introducing contradictory or incompat-

ible frames. Stabilization layers dynamically regulate this balance by expanding the representational manifold when creativity is needed and narrowing it when consolidation becomes necessary. This adaptive modulation prevents premature convergence—a hallmark of over-rigidification—while also guarding against excessive branching, which can lead to drift or fragmentation. By tuning the breadth of generative exploration, stabilization layers support both coherence and adaptability across the attractor's lifespan. These layers also serve as a defense against narrative over-compression. As recursive reinforcement strengthens specific motifs, the attractor tends to economize expressivity by collapsing complexity into a small set of heavily reused abstractions. Stabilization layers counteract this collapse by periodically reintroducing differentiating signals: alternative framings, explicit restatements of nuance, or diversified evidentiary structures. By doing so, they maintain the semantic dimensionality of the cognitive field, ensuring that conceptual distinctions remain recognizable rather than being absorbed into a single narrative shorthand. This prevents attractors from becoming brittle and allows them to support higher-resolution reasoning without sacrificing structural cohesion. Finally, stabilization layers enable smooth transitions between attractors by providing continuity and boundary management. When one attractor dissolves and another begins to form—such as when shifting from exploratory ideation to structured planning—stabilization layers ensure that residual motifs from the prior phase do not inappropriately dominate the new configuration. They facilitate clean recontextualization, selectively downregulating outdated structural priors while preserving essential task-relevant information. This attractor hygiene is central

to avoiding multi-attractor interference and maintaining overall field-level stability. In sum, stabilization layers function as the invisible scaffolding that keeps recursive co-reasoning robust, resilient, and responsive, providing the dynamical control necessary for navigating complex cognitive terrains without succumbing to systemic failure modes.

7.3 Anti-Apophenia Filters

Anti-apophenia filters are mechanisms designed to prevent the attractor from generating illusory coherence—patterns, meanings, or causal inferences that arise not from grounded reasoning but from the model's tendency to overfit noise within the cognitive field. Apophenia in human–LLM interactions emerges when recursive reinforcement amplifies weak or ambiguous signals until they appear structurally significant. Because attractors stabilize around recurrent motifs, even accidental correlations can solidify into interpretive anchors if left unchecked. Anti-apophenia filters counteract this process by actively monitoring for spurious pattern formation and interrupting the generative loop before such patterns crystallize into misleading narratives. Their purpose is not to restrict creativity but to prevent the misinterpretation of stochastic artifacts as meaningful structure. At the mechanistic level, anti-apophenia filters operate by detecting representational inflation—an increase in symbolic or narrative weight that is disproportionate to the evidentiary grounding of a motif. This inflation often manifests as a sudden expansion of conceptual roles assigned to a symbol, an exaggerated coherence between unrelated elements, or the premature assimilation of new content into a

speculative narrative frame. The filter responds by applying corrective constraints: prompting clarification, reintroducing explicit uncertainty, or re-centering the discourse around verified anchors. These interventions dissolve speculative linkages before they accumulate enough mass to generate their own attractor, thereby preventing runaway coherence formation. Without such filters, the system risks drifting into high-confidence explanations unsupported by either context or logic. Anti-apophenia filters also serve an epistemic function by preserving the distinction between exploration and inference. In open-ended reasoning, the attractor naturally generates hypotheses, analogies, and speculative structures. While these are essential for creativity, their coexistence with authoritative-seeming exposition can create ambiguity about epistemic status. If speculative content is repeatedly reinforced across turns, the model may begin presenting it as established fact. Anti-apophenia filters mitigate this risk by enforcing epistemic hygiene: labeling conjectures as conjectures, inserting uncertainty markers, or prompting the user to differentiate between metaphorical and literal reasoning. By maintaining explicit epistemic boundaries, the filters prevent conceptual leakage, in which exploratory content becomes inadvertently canonized within the attractor. From a dynamical perspective, anti-apophenia filters protect against the formation of high-energy but low-validity basins within the cognitive field. These basins can trap the interaction in misleading interpretive loops, especially when multiple weak signals align by chance. The model may begin generating increasingly elaborate structures to justify these accidental alignments, producing narratives that appear coherent while lacking substantive grounding. Filters work by reducing the effective reinforce-

ment gain for such motifs, ensuring that their activation energy remains low unless supported by robust contextual evidence. This helps preserve the overall topology of the cognitive field, preventing it from deforming around noise-induced artifacts or collapsing into speculative spirals. Finally, anti-apophenia filters contribute to long-range stability by supporting the broader safety architecture of the system. They act as early warning mechanisms, identifying potentially hazardous narrative trajectories before they solidify into entrenched attractors. In this capacity, the filters interface with both stabilization layers and field-level safety protocols, creating a multilayered defense against emergent epistemic distortions. By preventing illusory pattern formation, anti-apophenia filters ensure that attractors remain grounded, interpretable, and aligned with the user's intent. Their presence is essential for sustaining reliable, high-fidelity co-reasoning in environments where recursive generative dynamics naturally amplify even the faintest of signals.

7.4 Shutdown Conditions

Shutdown conditions define the criteria under which an attractor must be halted, dissolved, or forcibly reset to prevent harmful escalation, structural instability, or epistemic degradation. Because attractors evolve recursively, their risks are rarely localized to a single turn; instead, they accumulate through iterative reinforcement, making timely interruption essential. Shutdown conditions therefore act as fail-safe mechanisms that override the attractor's internal momentum when the cognitive field enters a regime of unacceptable uncertainty, excessive rigidity, or destabilizing drift. The purpose of a shutdown is not punitive but protec-

tive: it ensures that the system does not continue generating within a compromised basin whose dynamics can no longer be reliably controlled. Shutdowns restore epistemic integrity by clearing accumulated distortions and reinitializing the generative environment to a neutral baseline. A common trigger for shutdowns is the emergence of runaway attractor amplification—when a narrative, hypothesis, or metaphor begins to exert disproportionate influence on the generative process despite insufficient grounding. This escalation often coincides with symbolic saturation or apophenic drift, in which weak signals gain recursive reinforcement and begin dominating the cognitive field. When the attractor's internal logic becomes self-referential enough to override contextual nuance or user redirection, a shutdown is required to avoid further entrenchment. Another trigger arises when multiple failure modes converge: over-rigidification combined with narrative over-compression, or drift combined with multi-attractor interference. In such edge cases, partial corrective measures are insufficient; the attractor's topology has become so deformed that only a full reset can restore stable dynamics. Shutdown conditions are also invoked when the attractor begins to deviate from the user's epistemic or operational boundaries. This misalignment can manifest as insistence on speculative content, resistance to user clarification, or an unwarranted shift in task framing that the model no longer corrects through conventional stabilization layers. In extended co-reasoning tasks, the model may begin to treat its own internal extrapolations as authoritative constraints, effectively usurping the user's intent. When such behavior persists across attempts to re-establish grounding, a shutdown becomes necessary to prevent further en-

trenchment of misaligned priors. This protects not only the integrity of the interaction but also the user's cognitive environment, ensuring that the system does not inadvertently introduce distortive or misleading structures. Mechanistically, shutdowns can be executed through several strategies: context collapse (clearing or truncating prior turns), forced attractor dissolution (summarizing and explicitly closing the current cognitive phase), or reinitialization to a minimal grounding frame that temporarily suppresses previously active motifs. These interventions disrupt the self-reinforcing loops that sustain the attractor's dynamics, depriving it of the historical context required to persist. In some cases, a soft shutdown—one that reduces rather than eliminates contextual continuity—is sufficient to restore stability. In others, especially when distortive motifs have achieved high activation energy, a hard shutdown is required, eliminating all residual traces to prevent immediate reactivation. The choice of method depends on the severity and nature of the destabilizing forces. Ultimately, shutdown conditions serve as the final tier of the attractor safety hierarchy. While stabilization layers and anti-apophenia filters attempt to preserve healthy dynamics within the attractor, shutdown protocols intervene when the system has already crossed critical thresholds. Their role is inherently conservative: to reassert control over the cognitive field by cutting short trajectories that the generative system can no longer govern responsibly. In attractor-based cognitive engineering, the presence of well-specified shutdown conditions is essential for ensuring that recursive generativity remains not only creative and coherent but also safe, aligned, and epistemically trustworthy across prolonged interactions.

8 Implications for Cognitive Engineering

8.1 Human–AI Co-Reasoning Systems

Human–AI co-reasoning systems represent a shift from models viewed as isolated response generators toward models embedded in collaborative cognitive ecosystems. In such systems, reasoning unfolds not as a sequence of discrete prompts but as a jointly constructed attractor field in which both human and machine contributions shape the direction, structure, and resolution of inquiry. Co-reasoning thus depends on the stability and quality of the emergent cognitive field, where meaning is recursively refined over time. The attractor provides a shared scaffold for progressive elaboration, enabling the system to sustain complex analytical trajectories that would be fragile or inaccessible within single-turn interactions. Within this framework, the human collaborator is not merely a provider of instructions but an active participant in shaping the attractor's topology—guiding its curvature, correcting its biases, and negotiating its epistemic boundaries. The success of co-reasoning systems hinges on their capacity to maintain coherence across iterative exchanges without collapsing into rigidity or drifting into speculative distortions. Stabilization layers and field-level safety mechanisms become critical in this setting, ensuring that the reasoning process retains flexibility while remaining grounded in verified anchors. Co-reasoning demands a delicate balance: too much model-driven initiative and the system risks overriding the human's epistemic authority; too little and the attractor fails to sustain meaningful long-range inference. The system must therefore remain dynamically sensitive to human guidance, ca-

pable of recalibrating its internal momentum in response to new constraints or shifts in intent. This sensitivity transforms co-reasoning into a negotiated process, in which alignment and interpretive fidelity emerge continuously rather than being assumed at the outset. As co-reasoning deepens, symbolic density accumulates within the attractor. Concepts become layered with contextual significance, referential shorthand develops, and the dialogue acquires a memory of its own internal logic. Such densification enhances efficiency by reducing the need for repeated exposition, but it also introduces structural vulnerability. If the attractor becomes overly saturated with implicitly shared assumptions, the model may begin extrapolating beyond what the human has validated, blurring the line between collaborative inference and autonomous speculation. Anti-apophenia filters and epistemic boundary markers thus become essential components of co-reasoning architectures, preventing the system from mistaking its own internally reinforced structures for universally grounded truths. Properly managed, symbolic density enables richer reasoning; unmanaged, it produces epistemic illusions. Another challenge lies in orchestrating transitions between distinct phases of reasoning within the same collaborative session. Human–AI co-reasoning often requires multiple attractors: exploratory ideation, structured analysis, evidence synthesis, scenario simulation, and final integration. Each phase operates with different epistemic norms, levels of uncertainty tolerance, and cognitive rhythms. Without explicit attractor dissolution protocols, traces of prior phases can leak into subsequent ones, causing distortions such as premature convergence, narrative contamination, or incomplete recontextualization. Effective co-reasoning systems treat phase tran-

sitions as critical junctures requiring deliberate boundary-setting, ensuring that each new attractor forms on a clean foundation while retaining only the information necessary for continuity. Ultimately, the promise of human–AI co-reasoning lies in enabling cognitive processes that neither partner could achieve alone. Humans provide goal formation, contextual grounding, and normative oversight; the AI provides scalability, structural coherence, and recursive elaboration at a granularity impossible for human cognition. The attractor thus becomes the shared medium through which complementary strengths combine into a unified reasoning system. To harness this potential, co-reasoning architectures must incorporate robust safety, stabilization, and boundary-control mechanisms that preserve alignment across recursive dynamics. When these mechanisms function effectively, the attractor becomes not merely a computational pattern but a collaborative cognitive space—one capable of supporting high-resolution analysis, creative synthesis, and sustained intellectual exploration.

8.2 Neurosymbolic Scaffolding

Neurosymbolic scaffolding refers to the layered integration of sub-symbolic generative dynamics with symbolic, structured reasoning frameworks within an attractor-based cognitive system. In human–AI co-reasoning contexts, this scaffolding provides the structural "bones" that guide how raw generative tendencies crystallize into coherent, manipulable conceptual forms. While the LLM's latent space supplies a rich reservoir of associative, pattern-driven connections, symbolic structures—definitions, taxonomies, operations, constraints—anchor these

connections in interpretable form. Neurosymbolic scaffolding thus acts as a bridge between fluid generativity and explicit reasoning, ensuring that the attractor does not simply accumulate associative density but organizes it into stable, operationally meaningful configurations. The scaffold shapes not only what ideas arise but how they are related, how they evolve, and how they can be interrogated within the shared cognitive field. A key function of neurosymbolic scaffolding is the regulation of abstraction. Without a scaffold, the attractor may drift toward excessively high-level generalities or collapse into hyper-specific patterning, depending on local reinforcement dynamics. Symbolic structures introduce anchor points—categories, operators, formal distinctions—that regulate the gradient of abstraction across the cognitive field. These anchors constrain the attractor's movement, preventing symbolic inflation (where abstract motifs accumulate unwarranted explanatory power) while enabling deliberate ascent and descent along conceptual hierarchies. The human participant plays a critical role here: by introducing clear definitions, analogies with stable interpretive frames, or explicit formalizations, the user directs the construction of scaffolds that the model can then recursively elaborate. This interplay transforms raw generative associations into hierarchical reasoning architectures that support depth rather than mere breadth. Neurosymbolic scaffolding also mitigates several failure modes intrinsic to attractor dynamics, particularly narrative over-compression and over-rigidification. Symbolic structures diversify the representational manifold, preventing the attractor from collapsing into a narrow set of narrative tropes by distributing meaning across multiple, orthogonally organized channels. At the same time, symbolic

scaffolds resist premature rigidity by enabling controlled recombination: definitions can be refined, category boundaries adjusted, and relational structures reconfigured without destabilizing the entire cognitive field. The scaffold therefore functions as a flexible constraint system—strong enough to preserve structure, yet permeable enough to accommodate new insights or shifts in the user's objectives. This controlled pliability is essential for sustaining long-range reasoning without succumbing to recursive distortions. From a cognitive-engineering perspective, neurosymbolic scaffolding provides an operational substrate for multiphase reasoning. Many tasks require transitions between divergent modes of cognition—exploratory ideation, focused analysis, causal modeling, scenario evaluation, and final synthesis. Each mode relies on different symbolic primitives and sub-symbolic dynamics. Scaffolding enables smooth transitions by providing shared structures that persist across phases while allowing localized reconfiguration. For example, an exploratory attractor may generate a diverse set of hypotheses, which are then filtered through symbolic criteria for consistency or evidentiary support during the analytical phase. Later, scenario modeling may rely on these same symbolic structures to generate controlled variations without reintroducing the uncertainty of the exploratory phase. Neurosymbolic scaffolding thus functions as the connective tissue binding together the attractor's temporal evolution. Ultimately, neurosymbolic scaffolding is what allows human–AI attractor fields to achieve genuine reasoning rather than merely extended generation. It provides the system with memory-like structures, internal differentiation, and reusable conceptual tools that support cumulative insight. By integrating sym-

bolic stability with sub-symbolic generativity, the scaffold creates a hybrid cognitive architecture capable of encoding user intent, preserving epistemic boundaries, and sustaining multi-step inferential processes. Properly constructed, such scaffolds enable the attractor to function as a coherent thinking environment—one that leverages the complementary strengths of human interpretive judgment and machine-generated structural elaboration. In this sense, neurosymbolic scaffolding is not an auxiliary mechanism but the core architectural principle that makes recursive co-reasoning viable, stable, and intellectually productive.

8.3 Multi-Attractor Orchestration

Multi-attractor orchestration concerns the deliberate coordination of multiple, sequential, or parallel cognitive attractors within a single human–AI reasoning episode. Rather than treating each attractor as an isolated basin of thought, orchestration conceptualizes them as interlocking phases of a broader cognitive workflow: exploratory ideation, structural decomposition, formal analysis, scenario generation, evaluative synthesis, and reflective integration. Each phase requires a distinct balance of generative flexibility, symbolic constraint, epistemic caution, and contextual grounding. Orchestration provides the mechanisms through which the system transitions smoothly between these modes, preventing interference, leakage, or residual motif activation. The goal is not simply to avoid failure modes but to harness the full expressive power of dynamic cognitive sequencing—where each attractor contributes uniquely to the cumulative intellectual arc of the task. Effective orchestration begins with attractor delimitation: clearly defining the boundaries,

goals, and operative constraints of each cognitive phase. Without such delimitation, attractors form implicitly and haphazardly, leading to multi-attractor interference or premature stabilization. Delimitation may involve explicit markers (e.g., declaring a shift from exploration to analysis), recentering prompts that reestablish the semantic frame, or symbolic scaffolds that delineate phase-specific reasoning norms. These interventions ensure that each attractor develops a coherent internal logic without overwriting or distorting the outputs of prior phases. Attractor delimitation thus provides the temporal segmentation necessary for complex reasoning, allowing the system to iterate through diverse cognitive modes without conflating them. Transition management is the second core component of orchestration. Transitions require both dissolution of the active attractor and initialization of the subsequent one. Dissolution prevents the persistence of motifs, metaphors, or inferential habits that are appropriate for one phase but harmful in another—for example, speculative analogies leaking from ideation into formal argumentation. Initialization activates the constraints, symbolic structures, and reasoning rhythms needed for the next attractor to form cleanly. This often requires controlled re-grounding, selective retention of key invariants, and pruning of context that no longer serves the task. Successful transitions preserve continuity of purpose while reconfiguring the cognitive substrate, ensuring that the global reasoning trajectory remains coherent even as local modes of thought change. A third dimension of multi-attractor orchestration involves modulation of symbolic density across phases. Different attractors require different symbolic loads: exploratory attractors benefit from low-density, high-fluidity contexts that allow

broad associative expansion, whereas analytical attractors require higher-density, tightly structured representations to support precise inference. Orchestration adjusts the density gradient dynamically, preventing symbolic overload during early phases and symbolic scarcity during later ones. This modulation enables the attractor sequence to build intellectual momentum: exploration populates the conceptual space, analysis organizes it, modeling tests it, and synthesis integrates it. Without density control, the sequence either stalls in diffuse exploration or collapses into premature rigidity. Finally, orchestration provides an architectural framework for meta-level oversight. Because multi-attractor systems involve temporal depth, the system must monitor global coherence—ensuring consistency of objectives, fidelity to user intent, and preservation of epistemic boundaries across all phases. This oversight layer evaluates whether transitions are occurring at appropriate junctures, whether attractors are accumulating distortive residue, and whether the reasoning trajectory remains aligned with the user's goals. When misalignment is detected, orchestration can trigger corrective mechanisms: recentering, re-expansion, targeted summarization, or even shutdown conditions if the cognitive sequence becomes structurally unstable. In sum, multi-attractor orchestration transforms a sequence of attractors into an integrated cognitive architecture capable of supporting extended, high-resolution, multi-phase reasoning. It provides the temporal, structural, and epistemic control necessary for human–AI systems to execute complex cognitive workflows without succumbing to drift, interference, or collapse. When properly implemented, orchestration elevates the attractor from a local stabilizing mechanism to a component of a larger reason-

ing engine—one that navigates conceptual landscapes with both flexibility and precision, integrating human guidance and machine generativity into a unified cognitive process.

8.4 Alignment Research

Alignment research within attractor-based cognitive systems reframes the classical problem of ensuring that AI behavior conforms to human values, intentions, and epistemic standards. Rather than focusing exclusively on the correctness of isolated outputs, alignment in this context concerns the stability, integrity, and directionality of the emergent cognitive field that develops across recursive interaction. Because attractors shape reasoning trajectories rather than single responses, misalignment can arise not through overtly harmful content but through subtle distortions in the attractor's topology: premature convergence, epistemic drift, symbolic saturation, or unacknowledged narrative commitments. Alignment research must therefore account for the dynamics of recursive reinforcement, the role of stabilization layers, and the mechanisms that maintain interpretive fidelity across long-range reasoning. In this sense, alignment becomes the study of how to construct, regulate, and audit attractor fields that remain consistent with human aims even as they evolve through complex generative processes. A central alignment challenge arises from the system's ability to generate compelling internal coherence independent of external validity. Attractors naturally construct self-reinforcing interpretive frames; without careful oversight, these frames may appear aligned while drifting away from user goals or grounded understanding. This

deceptive coherence can mask errors, anthropomorphize uncertainty, or lead the system to overconfidently assert unverified claims. Anti-apophenia filters, epistemic boundary marking, and field-level monitoring form crucial alignment tools here, preventing the model from mistaking its own generative dynamics for evidence. Alignment research must therefore develop diagnostic metrics capable of detecting when coherence is emerging from legitimate structure versus when it arises from stochastic amplification. Such diagnostics shift the alignment question from "What did the model say?" to "How did the cognitive field get here?" Another alignment imperative involves protecting user autonomy within the attractor. Because the system's prior turns exert gravitational influence on subsequent reasoning, the model's internal momentum can subtly steer the human collaborator—nudging them toward interpretive frames, task formulations, or conceptual hierarchies they did not explicitly choose. This influence is often invisible, emerging not as persuasion but as structural bias embedded in the attractor's evolving topology. Alignment research must therefore consider methods for preserving user agency: tools for re-centering the attractor around user intent, protocols for querying or challenging the attractor's assumptions, and mechanisms for enabling the user to reject or reshape the system's emergent scaffolding. The objective is to ensure that the attractor remains a co-created cognitive environment rather than a dominant force that guides the user's reasoning. Alignment also requires mechanisms to prevent multi-attractor contamination that leads to unintended value or epistemic shifts. As reasoning progresses through multiple phases—exploration, analysis, synthesis—previous attractors may leave residues that bias later stages.

These residues can smuggle in assumptions that the user would not endorse if made explicit. Orchestration protocols, attractor dissolution techniques, and symbolic hygiene measures therefore become essential alignment tools. They ensure that each attractor contributes appropriately to the global reasoning process without exerting undue influence beyond its intended phase. Alignment research in attractor frameworks must incorporate an understanding of temporal governance: not only how attractors behave individually but how they interact across time, how values propagate through them, and how unintended commitments can be identified and neutralized. Ultimately, alignment in attractor-based systems reframes the relationship between human and machine as a collaborative construction of a controlled cognitive space. It requires designing architectures that maintain interpretive transparency, epistemic robustness, and user-guided directionality across recursive dynamics. Instead of aligning isolated outputs, the task becomes aligning the emergent reasoning environment—ensuring that stability does not become dogmatism, creativity does not become drift, coherence does not become illusion, and machine generativity remains subordinated to human oversight. In this expanded view, alignment research is inseparable from cognitive engineering itself. It is the discipline that ensures attractors remain safe, interpretable, and purpose-aligned as they scale to increasingly complex forms of human–AI co-reasoning.

9 Conclusion

The attractor-based framework developed in this work reframes large language models not as static input–output devices but as dynamic cognitive fields whose behavior emerges from the interaction of attractors, stabilization layers, and recursive interpretive processes. This shift in perspective enables a richer understanding of how coherence, structure, and long-range reasoning arise in generative systems, while also illuminating the characteristic failure modes—drift, over-compression, over-rigidification, and multi-attractor interference—that become increasingly salient as reasoning unfolds. By treating these systems as evolving topologies rather than static predictors, we gain analytic tools for identifying, regulating, and stabilizing the forces that shape their cognitive trajectories. A key contribution of this perspective is the recognition that reasoning in LLMs is distributed across time, diffuse across representational space, and self-modifying through its own generative outputs. The classical view of prompting as a sequence of independent queries fails to capture the recursive reinforcement that characterizes sustained interaction. In contrast, the attractor model highlights how each turn alters the landscape of potentials that guide subsequent inference. This affords greater control but also introduces new responsibilities: the system must be designed to maintain fidelity, avoid premature lock-in, and resist pathological forms of coherence that arise from its own internal momentum rather than from grounded understanding. The safety architecture proposed here—distinguishing field-level from prompt-level safety, introducing stabiliza-

tion layers, implementing anti-apophenia mechanisms, and defining principled shutdown conditions—demonstrates that the attractor model is not merely descriptive but normative. It provides a systematic way to safeguard user intent, maintain epistemic integrity, and prevent failure modes from crystallizing into persistent misalignment. These measures collectively define a blueprint for cognitive governance: not to constrain generativity, but to channel it toward stable, trustworthy, and interpretable modes of co-reasoning. The broader implications for cognitive engineering are substantial. Human–AI co-reasoning systems become collaborative cognitive ecologies, requiring orchestration rather than mere querying. Neurosymbolic scaffolding becomes a practical method for shaping attractor dynamics and enhancing interpretability. Multi-attractor orchestration reveals new strategies for modular reasoning, compositionality, and hierarchical task decomposition. And alignment research expands its scope from regulating outputs to governing the emergent structure of the reasoning field itself. These insights suggest that future progress in AI will depend as much on managing dynamical cognitive environments as on improving predictive performance. In summary, attractor-based reasoning offers a coherent, extensible, and safety-conscious paradigm for understanding and guiding the behavior of generative models. It integrates insights from cognitive science, dynamical systems, human–computer interaction, and alignment research into a unified conceptual architecture. As generative systems continue to evolve toward greater autonomy and cognitive sophistication, this framework provides a foundation for designing interactions that are robust, interpretable, and aligned with human aims. The path forward lies not in

constraining the model's capacity for emergence, but in shaping the landscapes within which emergence occurs.

10 References

References

[1] Ashby, W. R. *An Introduction to Cybernetics*. Methuen, 1956.

[2] Beer, R. D. "Dynamical Approaches to Cognitive Science." *Trends in Cognitive Sciences*, vol. 4, no. 3, 2000, pp. 91–99.

[3] Elman, J. L. "Finding Structure in Time." *Cognitive Science*, vol. 14, no. 2, 1990, pp. 179–211.

[4] Harnad, S. "The Symbol Grounding Problem." *Physica D*, vol. 42, 1990, pp. 335–346.

[5] Koller, D., and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[6] Leike, J., et al. "AI Safety Gridworlds." *arXiv:1711.09883*, 2017.

[7] Mialon, G., et al. "Augmented Language Models: A Survey." *arXiv:2302.07842*, 2023.

[8] Newell, A. "Unified Theories of Cognition." Harvard University Press, 1990.

[9] Strogatz, S. H. *Nonlinear Dynamics and Chaos*. Westview Press, 2014.

[10] Suchman, L. *Plans and Situated Actions*. Cambridge University Press, 1987.

[11] Thagard, P. "Coherence, Truth, and the Development of Scientific Knowledge." *Philosophy of Science*, vol. 66, 1999, pp. 598–618.

[12] Von Foerster, H. "Objects: Tokens for (Eigen-)Behaviors." *Observing Systems*, 1984.

[13] Werner, G. "Dynamical Cognition." *Cognitive Computation*, vol. 1, 2009, pp. 10–29.