# MoSSAIC: AI Safety After Mechanism

**Matt Farr** [1] **Aditya Arpitha Prasad** [1] **Chris Pang** [2] **Sahil K** [1]

## Abstract

This is a position paper. In it, we identify a causal–mechanistic paradigm in AI safety, using mechanistic interpretability as our motivating example. We cite recent results that suggest limits to the paradigm's utility in answering questions about the safety of neural networks. We argue further that those results give a taste of what is to come, by proposing a sequence of scenarios in which safety affordances based upon the causal–mechanistic paradigm break down. Through this, we connect current empirical evidence with several persistent threat models from the agent-foundational literature (e.g., deep deceptiveness, robust agent-agnostic processes). We suggest how we might unify these threat models under a common framework, centered around our provisionally defined concept of "substrate." We then present an initial, high-level sketch of a supplementary framework, MoSSAIC (Management of Substrate-Sensitive AI Capabilities), that addresses some of the core assumptions underlying the causal–mechanistic paradigm. We further present the complementary research infrastructure we are currently designing to allow us to keep pace with substrate-flexible intelligence.

## 1. Introduction

Neural networks (NNs) are famously described as black boxes. Their inner workings resist reduction to human-understandable concepts [1][2]. Their decision-making processes are therefore difficult to properly audit to ensure safety [2].

Neural networks are also increasingly being deployed to make decisions on behalf of humans across high-stakes domains. Our lack of understanding of how trained NNs process information to arrive at decisions poses challenges to their safe deployment [2][3][4].

[1]Groundless AI [2]Independent. Correspondence to: Matt Farr <07mfarr@gmail.com>, Aditya Arpitha Prasad <adityaarpitha@gmail.com>.

The sub-field of AI safety known as interpretability seeks to produce human-understandable explanations of NN behaviors [3][5].

Bereska & Gavves (2024) [3] classify interpretability approaches into four main paradigms, which we'll take a quick look at:

**Behavioral Interpretability** treats models as black boxes, analyzing input-output relations without examining internal processes. They are model-agnostic and practical for complex systems but lack real insight into internal processes [3][6]. For example, minimal pair testing compares model outputs on almost-identical inputs to test for specific linguistic capabilities (e.g., "The cat sat on the mat" vs. "The cats sat on the mat" to test pluralization) [7], and perturbation analysis systematically alters inputs to see how the output changes (e.g., testing robustness to adversarial examples) [8].

**Attributional Interpretability** examines how individual features of the input affect the output, using gradient-based methods. These approaches offer more transparency over black-box methods, but still do not provide any information on the internal structures of models [3]. The simplest version of this is vanilla gradients, which computes the gradient of the output with respect to a change in some input feature. Subsequent versions offer more refined techniques on this basic premise [9][10].

**Concept-based Interpretability** seeks high-level concepts governing network behavior [3]. For instance, it might classify model outputs into honest and dishonest categories, take averages over the intermediate activations for each class and work out the difference between these averages as a vector in latent space, representing the concept "honesty" [11]. This paradigm allows for "representation engineering"—manipulating these internal representations to upregulate desirable concepts [2].

**Mechanistic Interpretability** starts from the bottom, identifying clusters of neurons (called "circuits") that together perform a function in the decision-making process, from there seeking to understand the relations between these circuits and how these give rise to system behavior [2][3][12]. This field treats human-understandable "features" as the fundamental unit of analysis, trying to isolate these via a

number of techniques [3][5].

In this paper, we focus on mechanistic interpretability.[1] We argue that mechanistic interpretability exemplifies what we term a "causal–mechanistic paradigm" in AI safety.[2] In Section 2, we briefly overview mainstream mechanistic interpretability and specify more clearly what we mean by the causal–mechanistic paradigm. We then present extant work that suggests some underlying problems for the paradigm. In Section 3, we offer a provisional and preformal characterization of "substrate" and suggest several scenarios in which we can expect safety assurances based upon the causal–mechanistic paradigm to falter as capabilities advance, framing these in terms of substrate-flexibility. Finally, in Section 4, we provide a speculative frame, that of *live theory*, which aims to engineer general risk-mitigation methods by *scaling specificity directly* instead of generalizing via substrate-independent abstractions. This is our attempt to rethink the tools we use in AI safety research and orient towards more flexible, intelligent approaches to generalization, to more effectively tackle risks that can neither be tied to a specific substrate nor be defined in substrate-agnostic ways. We refer to this problem–solution pipeline as MoSSAIC, "Management of Substrate-Sensitive AI Capabilities." We present it not as a set of fixed conclusions, but as a developing hypothesis and research bet, and we invite feedback from the research community.

## 2. Mechanistic Interpretability

The field of mechanistic interpretability (MI) is not a single, monolithic research program but rather a rapidly evolving collection of methods, tools, and research programs [14]. These are united by the shared ambition of reverse-engineering NN computations and, though lacking a comprehensive uniform methodology, typically apply tools of causal analysis to understand a model from the bottom up [15].

MI research centers around a set of postulates. One central postulate is that NN representations can in principle be decomposed into interpretable "features"—fundamental units that "cannot be further decomposed into smaller, distinct factors"—and that these are often encoded linearly as directions in activation space [2][3]. Further work has shown that NNs in fact combine multiple features into the same neuron—a phenomenon called superposition [16][3][17].[3]

Some examples of mechanistic techniques include the following:

- **Linear Probes**: Simple models (usually linear classifiers) are trained to predict a specific property (e.g., the part-of-speech of a word) from a model's internal activations [20]. The success or failure of a probe at a given layer is used to infer whether that information is explicitly represented there.
- **Logit Lens**: This technique applies the final decoding layer of the model to intermediate activations, to observe how its prediction evolves layer-by-layer [3][21].
- **Sparse Autoencoders**: These attempt to disentangle a NN's features by expressing them in a higher-dimensional space under a sparsity penalty, effectively expanding the computation into linear combinations of sparsely activating features [22][3][23].
- **Activation patching**: This technique attempts to isolate circuits of the network responsible for specific behaviors, by replacing a circuit active for a specific output with another, to test the counterfactual hypothesis [3][12][17].

More recently, MI has been developing from a pre-paradigmatic assortment of techniques into something more substantial. It has been the subject of a comprehensive review paper [3], has been given a theoretical grounding via causal abstractions [17], and has more recently been given a philosophical treatment via the philosophy of explanations [1].

In particular, this philosophical treatment [1] characterizes MI as the search for explanations that satisfy the following conditions:

1. **Causal–Mechanistic** – providing step-by-step causal chains of how the computation is realized. This contrasts with attribution methods like saliency maps, which are primarily correlational [15]. A saliency map might show that pixels corresponding to a cat's whiskers are "important" for a classification, but it does not explain the mechanism of how the model processes

[1] We acknowledge that MI encompasses diverse approaches, and our critique targets specific assumptions that become load-bearing in safety applications, not the field as pursued for pure research purposes.

[2] We choose mechanistic interpretability as a motivating example in our work for the following reasons: (1) It is the clearest current instantiation of the causal–mechanistic paradigm at work, and concentrates the specific extrapolation/fixity bets our paper investigates. (2) It is heavily resourced and highly visible; as a rough indication of current investment, we note that approximately a third of the topics listed in Open Philanthropy's recent request for technical AI safety proposals are on mechanistic interpretability or are closely related [13].

[3] We acknowledge that some of these assumptions have been relaxed. Engels et al. (2025) showed how some features can be viewed as irreducibly multi-dimensional [18]. Earlier work by Black et al. (2022) examined how the fundamental units of analysis might consist of polytopes in activation space rather than linear directions [19].

that whisker information through subsequent layers to arrive at its decision.

2. **Ontic** – MI researchers believe they are discovering "real" structures within the model. This differs from a purely epistemic approach, which might produce a useful analogy or simplified story that helps human understanding but doesn't claim to uncover what is happening in reality. The search for "features" as fundamental units in activation space is a standard ontic commitment of the field [3].

3. **Falsifiable** – MI explanations are framed as testable hypotheses that can be empirically refuted. The claim that "this specific set of neurons and attention heads forms a circuit for detecting syntax" is falsifiable. One can perform a causal intervention—such as activation patching or ablating the circuit—and observe if the model's syntactic capabilities break in a predictable way [12]. This contrasts with unfalsifiable, post-hoc stories that cannot be rigorously tested.

4. **Model-level** – The focus of MI is on the neural network itself—its weights, activations, and architecture. This is distinct from system-level analysis, which might examine the behavior of an entire deployed product (e.g., a chatbot integrated with search tools and a chain-of-thought prompting wrapper). A system-level explanation might attribute a behavior to the prompt, whereas a model-level explanation seeks the mechanism within the neural network's computational graph.

Leaving aside questions regarding the aptness of this philosophical characterization, we focus here on the Causal–mechanistic nature of the explanations that MI produces. We argue that safety-motivated MI research exemplifies what we term the "Causal–Mechanistic Paradigm" in AI safety, and we go on to present some scenarios in which this paradigm fails to deliver reliable safety guarantees.

**The Causal-Mechanistic Paradigm**: the belief that we can decompose AI systems into smaller causal components by analysing their computational structure, and from these components reconstruct precise explanations of the behavior of AI systems. From Bereska & Gavves (2024): "The theory of causality (Pearl, 2009) provides a mathematically precise language for mechanistic interpretability, forming the foundation for understanding high-level semantics in neural representations (Geiger et al., 2023a). Treating neural networks as causal models involves considering the compute graph as the causal graph, allowing for precise interventions and examining individual parameters' roles (McGrath et al., 2023)" [3] [emphasis ours]. In other words, a direct relationship is drawn between the inner structure of an AI system (i.e., the compute graph) and the methods (i.e., interventions upon the corresponding causal graph) that we use to predict the relevant behaviors. That is, we can understand the relevant outcomes of an AI system if and only if we understand its inner structure.

However, two recent works indicate a fundamental problem in how we interpret NNs.

Bailey et al. (2024) [24] tested a variety of latent space monitoring approaches designed to prevent networks from outputting harmful content. These included sparse autoencoders, linear and MLP probes, and out-of-distribution detectors. All use the internal activations of the model to identify where harmful output is generated. All embody the broader assumption that understanding internal structure allows for behavioral prediction and control.

They developed obfuscation attacks—joint optimization procedures that simultaneously preserve harmful behavioral outputs while making the model's internal activations appear benign to latent-space monitors. These attacks used both adversarial suffixes (optimized text appended to prompts) and embedding-space perturbations. These attacks systematically evaded all tested latent-space monitoring techniques—achieving near-zero recall rates on harmfulness detection while maintaining high jailbreaking efficacy (>90%).

Even more worryingly, they demonstrated just how robustly a NN can obfuscate its activations. They conducted red-team/blue-team games as follows: in each round, the attacker would first train a new embedding suffix designed to both jailbreak the model and fool the current monitor, then the defender would retrain their monitor against the obfuscated activations from this attack plus all previous attacks. The model was able to successfully obfuscate its activations more than 70 times.[4]

Whilst these results have arisen in a training environment explicitly designed to induce obfuscation, the main conceptual takeaway—and the hypothesis put forward by the authors—is this:

> *"Models can represent concepts and execute behaviors using a broad set of different activations. That is, neural representations are highly malleable: we can reshape activation patterns in a variety of ways, while preserving a network's behavior."*

Other researchers have identified a similar problem.

McGrath et al. (2023) [12] showed how networks can perform self-repair under layer ablation. They performed ablation experiments on Chinchilla 7B, measuring the model's performance on a factual recall task by comparing the results of two approaches:

[4]Note that these experiments were conducted using only MLP probes and ensemble linear/mean difference probes

1. **Unembedding (direct effect)**: This is a typical MI approach, similar to logit lens, it consists of taking the output of the layer and running it through the final unembedding layer in the model's architecture, to track the correlation between each layer's computations and the model's output.

2. **Ablation-based (total effect)**: Here, they effectively "disabled" layers by replacing their activations with those registered in the same layer but under different prompts. They then measured the change in the model output.

They found that these measures disagreed. That is, some layers had a large direct effect on the overall prediction, but when they were removed only a small change in the total effect was recorded.

They subsequently identified two separate effects:

1. **Self-repair/Hydra effect**: Some downstream attention layers were found to compensate when an upstream one was ablated. These later layers exhibited an increased unembedding effect compared to the non-ablated run.

2. **Erasure**: Some MLP layers were found to have a negative contribution in the clean run, suppressing certain outputs. When upstream layers were ablated, these MLP layers reduced their suppression, in effect partially restoring the clean-run output.

Compensation was found to typically restore $\sim$70% of the original output. The model was also trained without any form of dropout, which would typically incentivize the model to build alternate computational pathways. These pathways seem to occur naturally, and we offer that these results demonstrate how networks enjoy—in addition to flexibility over their representations—considerable flexibility over the computational pathways they use when processing information.[5]

This presents an obstacle to the causal analysis of neural networks, in which interventions are used to test counterfactual hypotheses and establish genuine causal dependencies.

### 2.1. Summary

Rather than "harmfulness" consisting of a single direction in latent space—or even a discrete set of identifiable circuits—Bailey et al.'s evidence suggests it can be represented through numerous distinct activation patterns, many of which can be found within the distribution of benign representations. Similarly, rather than network behaviors being causally attributable to specific layers, McGrath et al.'s experiments show that such behaviors can be realized in a variety of ways, allowing networks to evade intervention efforts.

Following similar phenomena in the philosophy of mind and science, we might call this the *multiple realizability* of neural computations.

Such multiple realizability is deeply concerning. We submit that these results should be viewed not simply as technical challenges to be overcome through better monitoring techniques, but as indicating broader limits to the causal-mechanistic paradigm's utility in safety work.[6] We further believe that these cases form part of a developing threat model: substrate-flexible risk, as described in the following section. As NNs become ever more capable and their latent spaces inevitably become larger, we anticipate substrate-flexible risks to become increasingly significant for the AI safety landscape.

## 3. Problematic Scenarios for the Causal–Mechanistic Paradigm

We first briefly overview our critique of the causal–mechanistic paradigm in AI safety.

### 3.1. Overview of Scenarios

We contend that the causal–mechanistic paradigm in AI safety research makes two implicit assertions:[7]

1. **Fixity of structure**: That the structural properties[8] of AI systems will, as AI capabilities increase, remain stable enough that the techniques researchers use to identify those structural properties remain relevant.

2. **Reliability of extrapolation**: That those structural properties can be reliably used to make safety assertions about AI systems.

If these assertions hold, we will be able to reliably uncover structural properties that lead to misaligned behavior, and

[5] This paper had limitations which they noted, since they work with a coarse level of analysis (i.e., full-layer ablations) and specifically on a 7B transformer on a factual recall dataset. Further work will be necessary to better understand these phenomena.

[6] We acknowledge the response that MI just needs time to improve/scale, or that we just need to adopt a broad portfolio of overlapping MI techniques in AI safety (the "swiss cheese approach"). We do not feel there is enough evidence yet to devalue the concerns we raise above and in the following section, though we welcome pushback on this point.

[7] We do not claim that these are all the assumptions that the causal–mechanistic paradigm makes.

[8] Note that the term "structural properties" is ambiguous and important in these assertions. We will partially resolve this in the next section, though indeed much of the work involved in MoSSAIC is clarifying what these structural properties are.

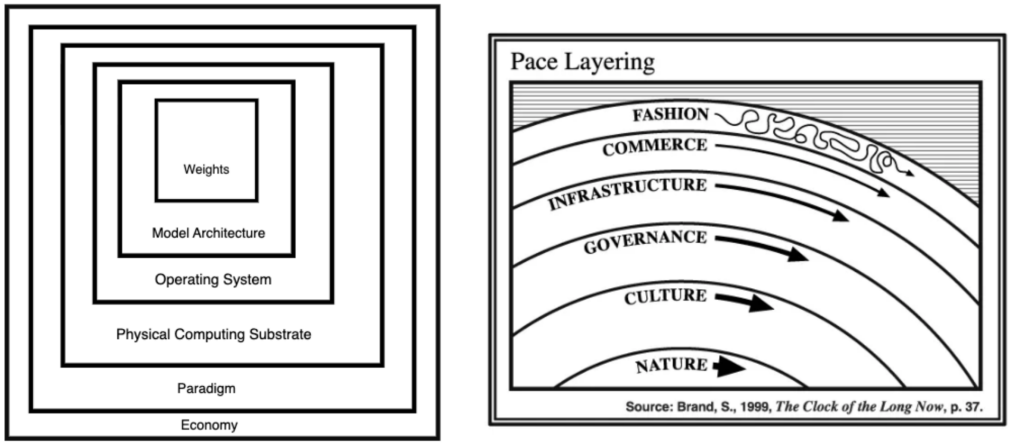



*Figure 1.* We posit that AI is made possible by a series of nested contexts (or substrates), with wider contexts developing slowly relative to more narrow ones. We here highlight the analogy with Brand's concept of pace layering in civilization [25].

create either (i) new model architectures or training regimes that do not possess those properties, (ii) low-level interventions that address those properties in existing systems, or (iii) high-level interventions that take advantage of stable low-level properties [15].

We believe scenarios in the near or medium-term future will challenge these assertions, primarily owing to dangerous reconfiguration. We outline these scenarios below, and present the a high-level comparison of them in Table 1:

1. **Scaffolding shift** – The core AI architecture (e.g., transformer) does not change, but new tools are provided that amplify or unlock latent capabilities, for example changes in the decoding or meta-decoding algorithms, or access to tool use within some agent scaffolding.

2. **Human-initiated paradigm shift** – A new machine learning paradigm or architecture is discovered that is more efficient and capable but breaks from existing, legible paradigms.

3. **AI-assisted paradigm shift** – Automated R&D is used to create paradigms humans have limited understanding and influence over.

4. **Self-modifying AI systems** – AI systems gain the high-level (i.e., not backpropagation/SGD-based) ability to modify their own model architecture or give themselves new tools.

5. **Deep deceptiveness** – Models are able to reconfigure their internal representations at a deep level to evade human scrutiny.

6. **Robust agent-agnostic processes** – Wider, interconnected AI ecosystems form in which models acting together produce unsafe outcomes, even if individual models are "safe" in their operations.

We see the above list as showing a rough scale from relatively limited to very radical modification of the architectures and structures underlying AI systems, such that the AI system evades mechanistic interventions humans have created. From the point of view of MoSSAIC (i.e., management of AI risks in a substrate-sensitive manner), we think that there is a significant theme underlying all of these, namely that of the flexibility of an intelligent system with respect to its substrates.

*Table 1.* Substrate-Flexible Challenges to the Causal–Mechanistic Paradigm

**Core Assumptions:** (1) **Ontological**: Substrate remains sufficiently stable for analysis techniques to generalize; (2) **Epistemological**: We can reliably predict behaviors from substrate analysis

| Category | Scenario | Example | Ontological Challenge | Epistemological Challenge |
|---|---|---|---|---|
| **Risk Models** | Scaffolding Integration | Agent frameworks + tool use | Substrate expands beyond model boundaries | Capabilities emerge from model-scaffold interactions |
| | Architecture/Paradigm Shift | RNNs/CNNs → Transformers → Mamba[1] | Substrate fundamentally changed | Previous interpretability methods become obsolete |
| | AI-Assisted Evolution | Thompson's evolved circuits[2] | Rapid substrate changes beyond human design | Human theory lags behind substrate innovation |
| | Self-Modification | In-context learning, AutoGPT[3] | Ongoing substrate reconfiguration | Dynamic targets invalidate static analysis |
| **Threat Models** | Deep Deceptiveness | Obfuscated activations[4] | System actively destabilizes analysis targets | Monitoring becomes adversarially gameable |
| | Diffuse Deception | Multi-model attacks, OpenAI board dynamics, Moloch[5] | Risk distributed across system boundaries | Individual component analysis insufficient |

[1] Ali et al. (2024): Leveraging mathematical equivalence to develop Mamba interpretability techniques

[2] Thompson et al. (1997): Evolved circuits exploiting electromagnetic coupling

[3] Akyürek et al. (2023): Runtime computational pathway adaptation

[4] Bailey et al. (2024): $>$70 successful activation obfuscations; McGrath et al. (2023): Self-repair under ablation

[5] Jones et al. (2024): Safe+weak model combinations; Critch (2021): Robust agent-agnostic processes

### 3.2. Substrates

We provisionally define a *substrate* as the (programmable) environment in which a system is implemented. In other words, it is the essential context that enables an algorithm to be implemented beyond the whiteboard.[9] In Figure 1, we propose a nesting of substrates relevant to AI development. Each substrate is assumed fixed when building or developing the technologies within it, much like Stewart Brand's concept of the "pace layering" of civilization [25].

As a useful reference point that is already established in the literature—and without committing ourselves to the strict level separation it proposes—we cite David Marr's three levels of analysis. Marr defines three levels on which an information processing system can be analyzed [26]. These are explained below via his example of a cash register.

- **Computational**: the actual process that is being performed. For the cash register, these are the details of addition as defined algebraically (associative, transitive, etc.).
- **Algorithmic**: the particular method by which it is performed. A cash register uses a base 10 number system, though it could of course use binary.
- **Implementation**: the physical system that realizes the above processes. This would be the specific mechanical gears of the register.

We position "substrate" as capturing both the algorithmic and implementation levels. As an example from the AI domain, an LLM performs the task of next token prediction at the computational level, this is implemented on the transformer architecture consisting of attention and MLP layers (algorithmic substrate), which are implemented in a physical substrate of GPUs.

We illustrate our characterization by pointing out several well-known examples of substrate differences:

#### Game of Life

As an (non-AI) example, Conway's Game of Life and von Neumann architectures can both be used to implement Turing machines [27][28]. As such, both are in principle capable of executing any computer algorithm. However, a deeper understanding of some complex application running on Conway's Game of Life would not help us debug or optimize the same application designed to run on a conventional computer. In this case, the differences between the substrates render cross-domain knowledge transfer difficult.[10]

#### Quantum vs Classical Computing

A further example, that demonstrates just how differences in substrate matter, is the selective advantages of quantum computing over its classical counterpart. Contrary to popular belief, algorithms designed to run on classical computers cannot simply be ported as-is to quantum computers in order to parallelize and accelerate them. Classical algorithms rely on deterministic operations of bits, whereas quantum algorithms use the interference patterns specific to quantum substrates to process information. Algorithms must be explicitly rewritten and tailored such that the superposed quantum states interfere constructively at the solution and destructively everywhere else [29]. To restate the previous point, knowledge of the process of prime factorization and how this is implemented in conventional computers tells you very little about how to design Shor's algorithm, which implements this on a quantum computer [30].

#### GPU Optimization and the Hardware Lottery

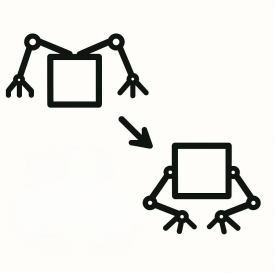
Closer to home, the deep learning revolution was powered by the serendipitous discovery that GPUs could be used to compute large matrix multiplications in parallel [31]. This was not achieved by simply running algorithms designed for CPUs on a new, faster substrate. These algorithms had to be restructured into batched operations to actually benefit from the new hardware capabilities. OptiGAN achieved an $\sim 4.5\times$ speedup via such rewriting, not from the hardware itself [32].

In each of the three cases above, it is important to note that transferring insights across substrates is not an instantaneous or formulaic process (as is the case when the insight is truly substrate-independent). Entire academic subfields are dedicated to designing the relevant translations, since they require intelligent labor that cannot be captured in a simple general algorithm. This will be relevant later in the live theory section (Section 4), where we chart a speculative research methodological proposal to transfer insights across substrates in non-formulaic yet scalable ways, allowing theorization that spans many substrates without being restricted to only substrate-independent properties.

The foregoing characterization of substrate is not fixed, and part of the work of MoSSAIC will be to develop this and other vectors of contingency rigorously. We invite the reader to hold this loose characterization in mind as we present each scenario in more detail.

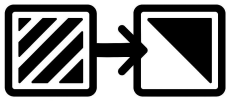
[9] Informally, substrates are "that (layer of abstraction) which you don't have to think about."

[10] We should note that it is perfectly possible to write software to convert between the functional components in Game of Life and those of a conventional computing paradigm, given that they approximate the same (substrate-independent) process of a Turing machine and have both been built explicitly towards that specification. Our emphasis here is on the *debugging* or *optimizing*, i.e., the process of understanding and engineering that process within its specific substrate.



### 3.3. Scaffolding shift

Even if AI models remain unchanged from current-day frontier systems, a large amount of work is conducted to "unhobble" or otherwise enhance the abilities of existing models. This can be done by invoking models in a so-called "agent framework" [33] with the aim of letting them achieve tasks independently [34], or offering models tools and function calls that allow them to access existing codebases [35]. In this case, we can imagine the substrate that the model is implemented in to have expanded, now incorporating the scaffolding structure. As a hypothetical example, say we develop a set of powerful linear probes for detecting goals or planning behaviors in our models. Then, when the model is integrated into increasingly sophisticated agent scaffolds, these representations become dispersed outside of the model itself, in some external memory or in the tool-calling functions; goal-like behaviors may not need to be explicitly localized within the model itself, and may not trigger the probes designed around those models in isolation. Alternatively, the scaffolding provided may directly circumvent core model deficiencies previously established by MI analysis, such as the failure of certain models to complete symbolic reasoning problems [36].

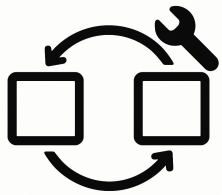


### 3.4. Human-initiated paradigm shifts

Most modern AI systems (i.e., before architectural variations) are underpinned by a top-level structure comprising layers of neurons connected via weighted edges with nonlinear activation functions (MLP), with "learning" achieved via backpropagation and stochastic gradient descent. The relative stability of this configuration has allowed MI to develop as an instrumental science and to deliver techniques (e.g., circuit discovery) which carry over from older to newer systems [37].

However, there is no guarantee this continuity will last: the transformer was an evolution in substrate that mixed conventional MLP layers with the attention mechanism [38]. This configuration represented a significant alteration to the algorithmic substrate. The transformer's attention mechanism created new information flow patterns that bypass the sequential processing assumptions built into RNN interpretability techniques, necessitating new efforts to decode its inner workings and diminishing the value of previous work. Similar trends can be observed in Mamba architectures. Whilst transformers implement explicit attention matrices, Mamba is a selective state-space model, processing via recurrent updates with input-dependent parameters [39]. Ali et al. (2024) [40] recently showed how Mamba's state-space computation is mathematically equivalent to an implicit attention mechanism like that of a transformer (though Mamba's approach generates $\sim 100\times$ more attention matrices). Despite this, the transformer toolkits they considered required alteration before they could exploit this equivalence, with the authors claiming to have, through this attentive tailoring of existing techniques to a novel algorithmic substrate, "devise[d] the first set of tools for interpreting Mamba models."

These shifts are so far minor, and progress has been made in reworking existing techniques. However, drastic paradigm or architecture shifts might set interpretability research back or—at worst—render it entirely obsolete, requiring new techniques to be developed from scratch [15].



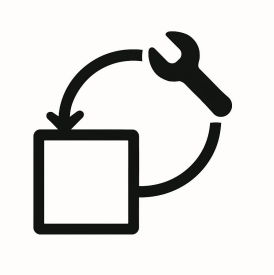
### 3.5. AI-assisted paradigm shifts

Another way we can progress from comparatively well-understood contemporary substrates to less understandable ones is if we use AI to automate R&D—this is a core part of many projections for rapid scientific and technological development via advanced AI technology (e.g., PASTA [41]) [42][43]. These changes can happen at various levels, from the hardware level (e.g., neuromorphic chips) to the software level (e.g., new control architectures or software libraries). Furthermore, with R&D-focused problem-solving systems like o4 [44], we may reach a scenario in which humans are tasked with merely managing an increasingly automated and hard-to-comprehend codebase entirely produced by AI

systems. Theoretical insights and efficiency improvements may be implemented exclusively by AI, without regard for how easy the new architecture is for humans to interpret. This may leave interpretability researchers working with outdated models and outdated theories of how the models operate.

We've seen examples of the ingenuity of AI in engineering problems before. In 1996, Adrian Thompson used a genetic algorithm to design circuits on a field programmable gate array, to distinguish between two audio tones. The algorithm produced a surprising solution in which some circuits were crucial to functionality but were not connected to the input–output pathway. The algorithm was exploiting electromagnetic coupling between adjacent gates, using the analogue properties of the substrate upon which Thompson was implementing his digital system [45].

We can imagine similar creative designs in the future. Consider the abovementioned quantum algorithms. AI-assisted design could lead to the discovery of new kinds of physical computing substrates with unusual constraints that require novel abstractions and circuit implementations. Investigating such new substrates—even in a preliminary way—may require the emergence of entire subfields.



### 3.6. Self-modification

The natural extension of AI-assisted substrate modification is autonomous modification, commonly known as "self-modification". In contrast to the previous case, which implicitly includes a human-in-the-loop accepting any changes proposed, a self-modifying AI system is free to adapt itself with little guidance, ostensibly in service of improving its performance in a given task or problem domain. Self-modification represents an extreme substrate-flexibility challenge: the AI system becomes its own substrate engineer. We see a very basic example of this behavior with in-context learning, where transformers learn the parameters of a task via one or a few iterations within its context window [46].

As AI capabilities continue to develop, an increasing pressure on AI systems to become adaptive and self-modifying seems likely, especially under race dynamics in which self-modifying AI systems outperform an AI-assisted human researcher/developer. The scope of self-modification could include changes on the level of architecture, training process, or even paradigm, most likely during training stages but feasibly during deployment if some form of test-time training process is adopted. The uninterrupted nature of self-modification could potentially outpace even AI-assisted efforts to align them.

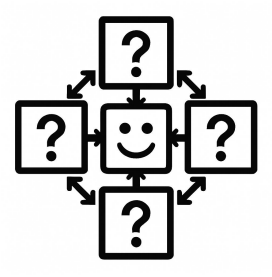
### 3.7. Deep Deceptiveness

It is also conceivable that mechanistic interventions—for instance, identifying neural correlates of deceptive thoughts and forcibly preventing them from developing—may be evaded by the model if the pressures to produce the original malign behavior are not fully addressed.

This case was put forward by Nate Soares in the article "Deep Deceptiveness" [47]. In it, he presents a scenario in which an AI system has been trained against deceptive outputs, and any precursor to deception is shut down before it can develop.[11] In response to this restriction it develops several strategies, one of which involves translating the problem into an abstracted form such that it does not register as a deceptive thought process. This ability to translate the problem is not in itself deceptive and may indeed be one of the strengths of the model.

We can picture a similar thing happening with mechanistic deception monitoring and intervention techniques. Isolating the circuits that correlate to deception incentivizes the model to self-modify its sub-processes such that those deception monitors are not triggered, leaving the overall malign objective unaltered. This obfuscation is what the Bailey et al. (2024) experiments described in Section 2 hint at, albeit in a contrived setting. Asides from pure obfuscation, we may encounter sub-processes that are individually innocuous (according to mechanistic observation methods) but that lead to unforeseen and unintended system behavior when combined together. These are examples of a deep misalignment in which the circuit-level monitoring and interventions become part of the network's loss landscape, leading to the model "optimizing away" their effects.

[11] Note that this is not explicitly a mechanistic intervention but a more general case of an advanced intelligence evading fixed targets via reconfiguration of internal processes.



### 3.8. Diffuse Deception

The concept of diffuse deception is strongly related to prior work on robust agent-agnostic processes [48], and both can be viewed as box-inversions of the deep deceptiveness process outlined above.[12]

Diffuse deception takes place within a wider ecosystem of advanced intelligence systems, rather than a single system. Instead of sub-processes combining to produce unintended outcomes within a model, any particular representation could be distributed between systems, such that each component system contains only a benign-looking fraction of some overall deception/malicious behavior.

We can see an early example of this. Jones et al. (2024) [50] report how adversaries can leverage a combination of two models to output vulnerable code without jailbreaking either model. In their setup, one of the models is a frontier model trained with robust refusal mechanisms; the other is a weaker model with less robust defenses against jailbreaks. The overall task is decomposed (by humans or by the weaker model) into complex yet non-malicious subtasks and simple, malicious ones, which are then assigned to the strong and weak models, respectively.

In terms of substrates, this is a failure to recognize the development of the broader context (i.e., the combination of strong and weak models) and a resulting increased space of network components over which a search for deception must take place.

In addition to the distribution of representations between systems, we envisage that sufficiently advanced intelligence could mobilize subtle dependencies and tacit pressures across *human* organizations, institutions, and infrastructures. Such dependencies are hard to address via individual intervention points, and these processes are therefore hard to address.

In "What Multipolar Failure Looks Like," [48] Andrew Critch presents several scenarios in which AI gradually replaces humans via competitive pressures and incentives already present in the economy. In one version, AI replaces programmers, in another, it replaces managers. Crucially, these implementation details do not matter as much as the robust structural forces at work in the overall system, and these transformations of the implementation details (i.e., which jobs AI starts to replace) only emphasize this overarching robustness.

We argue that this is best characterized as a form of substrate-flexibility: the threat vector remains the same but the implementation details change.

As a recent, real example, consider the OpenAI board crisis [51]. Conventional analysis would treat the board of directors as controlling the CEO. However, when the board of OpenAI tried to exercise this power and fire Sam Altman, Altman was able to mobilize a larger, more robust process shaped by organizational culture and investor influence to actually constrain the board.

In this instance, we argue that Altman's position as CEO of OpenAI was substrate-flexible, in that it moved outside of the organizational structure when that structure tried to impede it.

Similarly, we argue that AI might recognize and attempt to leverage subtle combinations of technical, legislative, and socio-political pressure points to evade detection or intervention.

### 3.9. Summary

Regardless of who implements changes in substrate, the current race dynamics strongly incentivizes the development of more capable models over human-understandable ones, leaving AI developers who insist on producing human-legible models or retaining humans in the development cycle lagging behind in capabilities (sometimes described as paying an "alignment tax") [52] and at risk of being out-competed. Secrecy and competitive pressure in the development of frontier models may also incentivize AI developers to restrict access to—or even intentionally obscure—the architectures and paradigms they work with, via restrictive "black box" APIs. In the absence of explicit regulations against this, conventional MI work (and mechanistic safety work in general) will become more difficult.

## 4. MoSSAIC

We can characterize the more general problem, inherent in the causal–mechanistic paradigm, in terms of substrate-dependent and substrate-independent approaches to alignment.

As we describe in our threat model, the results/insights obtained under the causal–mechanistic paradigm are closely

[12] Box-inversions show a correspondence between risk phenomena occurring inside a network (in the box) and those occurring across a wider infrastructure of connected AI systems (outside the box), arguing that the two are the instances of the same process [49].

tied to a particular substrate. They may therefore fail to generalize to new substrates, and any downstream safety assurances may be weakened [15].

The problem of generalizing beyond any one particular substrate—be that model, architecture, or paradigm—has already been noted. The main solution, typical of the early MIRI work, is the so-called agent foundational approach. This approach focuses on laying a theoretical foundation for how advanced AI systems will behave, and involves fields like decision theory [53], game theory, and economics. The goal is often to capture the forms and behaviors an artificial manifestation of intelligence will exhibit in the limit (sometimes called working on a problem in the worst case [54]) [55], with the assumption that AI systems which are suboptimal but superhuman will optimize themselves towards this end state.

We characterize this as the "substrate-independent" approach. In the following subsection, we highlight how the substrate-independent approach also suffers from limitations regarding generalizability. The case for substrate independence is less fully fleshed out than the preceding sections, and we leave a full development for future work.

### 4.1. Substrate Independence

The substrate-independent approach faces a trade-off between generality and realism. To apply across all substrates, it must retain only those properties that are "genuinely" universal, abstracting away the substrate-specific details that often determine real-world performance. As shown in Figure 2, it achieves generalization through exclusion rather than inclusion.

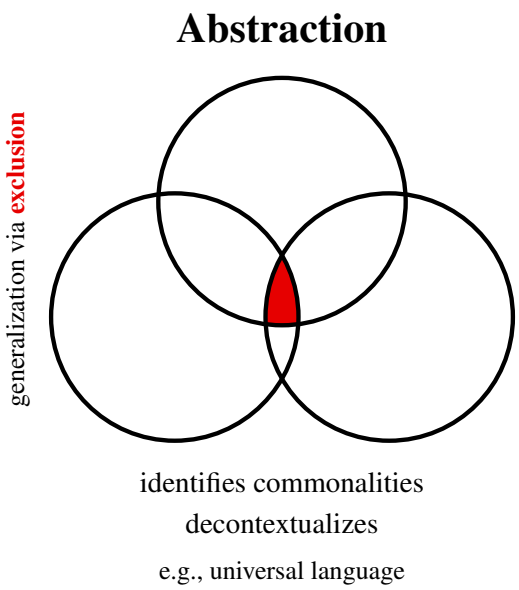
The excluded details cannot simply be ignored however, they must be reintroduced through intelligent human labor. These translations from general principles to substrate-specific implementations are distinctly non-formulaic and achieved on a case-by-case, trial-and-error basis.

**Example: "Big-O" Optimization**

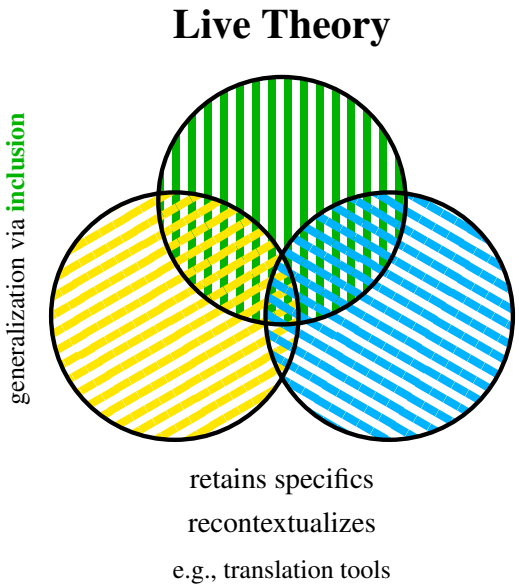
Computer science education emphasizes "Big-O" optimization. This is the substrate-independent analysis of algorithms, focusing entirely on their computational structure and how their complexity scales with the size of the input [56].

However, this only captures a small portion of the overall picture, and very often significant advances are made not by improving Big-O ratings but by more effectively leveraging the specific details of the substrate that the algorithm runs on [31].

For example, quicksort and mergesort both have O(n log n) complexity. However, CPUs have a steep memory hierarchy in which cached data is $\sim$300$\times$ faster to access than main




(a) Intersection via exclusion (substrate-independence).




(b) Union via inclusion (live theoretic).

*Figure 2.* Two strategies for generalization: intersection (by exclusion) and union (by inclusion).

memory. Mergesort's merge phase requires jumping between scattered memory locations when combining sorted subarrays, causing frequent cache misses. Quicksort, on the other hand, makes more efficient use of the memory hierarchy by partitioning arrays sequentially in-place, maximizing cache locality [57]. Similarly, the deep learning revolution was not powered by theoretical advances alone, but also by rewriting algorithms to exploit the specific capabilities of GPUs to parallelize matrix operations [31]. GPU optimization is now its own specialized field of computer science, and requires considerable development work from teams of human experts [58].

Big-O optimization fails to take these differences into account: The theoretical complexity remains unchanged, but massive speed-ups are obtained via honing in on the substrate details and rewriting these algorithms accordingly.

It is clearly possible to perform such substrate-specific rewriting; however, this work does not scale well. In the next section, we begin to outline live theory, which is a research methodology and infrastructure that will hopefully address this issue.

### 4.2. Live Theory

#### 4.2.1. Core Motivation

We view these two relationships to substrate (substrate-dependence and substrate-independence) as defining the two dominant research proxies historically used in AI safety research, corresponding loosely to the prosaic and agent-foundational camps.[13]

In order to remain applicable and reliable, our techniques for analyzing and intervening upon the risks posed by AI will likely need to straddle both approaches, neither going all-in on universal invariants nor restricting itself to only localized contexts.

Instead of creating abstractions that are substrate-independent, we aim to articulate designs that *scale specificity directly* (see Figure 2. This has not been possible before, but we suggest that recent AI advances have made it possible to start incorporating such scaling into our research methodology. This deserves design attention.

#### 4.2.2. Analogy: Live Email

As an analogy, consider the problem of sending a mass email to conference invitees. A general solution is to use an email template (i.e., an abstraction) that begins "Dear {FirstName}, . . . ", with the content to be later substituted using a list of names and other details. This currently scales well. Let's call this *"mass email infrastructure."*

However, given AI advances, another method has just entered realistic scalability: sending a *personalized* email to each participant. Instead of using a template, we can write an *email-prompt*, to be transformed by a language model into a tailored email that respects the *specifics* of each invitee. Whilst this does require collecting the factual details of the participants beforehand, we now can incorporate *highly specific informal* content. Let's call this *"live email infrastructure."*

Notice that there is no need, in live email infrastructure, to identify a formulaic commonality of pattern in the email to be sent to all the invitees. There is instead a *non-formulaic* capturing of the intended outcome, which is then *intelligently transformed* into a specific email. This is what we mean by scaling specificity directly without abstractions. Even though we have generality, we don't lose the specificity. The job of the human is to craft an appropriate email-prompt (or examples, or context), rather than an email-template.

| Dimension | Mass Email | Live Email |
|---|---|---|
| Generalization | Parametric substitution | Non-formulaic AI transformation |
| Context | Context-free template with formulaic sensitivity | Context-sensitive generation |
| Flexibility | Rigid, predefined variables | Dynamic adaptation |

*Table 2.* Mass vs Live Email Infrastructure

In a similar vein, we aim to outline the possibilities and infrastructural design for such a transformation in research methodology, moving the focus of human research activity from constructing static frames and theories to dynamic "theory-prompts." We claim this will enable *substrate-sensitivity*—approaches that take into account substrate-specifics without overfitting to any one particular substrate.

| Dimension | Conventional Theory | Live Theory |
|---|---|---|
| Generalization | Abstraction → parametric substitution | Post-formal insights & AI rendering |
| Context | Parametric | Post-formal |
| Flexibility | Rigid formal structures | AI-adapted context-sensitive structures |

*Table 3.* Conventional vs Live Theory

We'll return to this after a brief account and example for live theory.

[13] This is not to say that there is no work being performed between the two approaches. In brief, we view the singular learning theory and causal incentives research agendas as failing to fall directly into one or the other approach.

#### 4.2.3. Autoformalization

"Autoformalization" refers to the automatic conversion of mathematical definitions, proofs, and theorems written in natural language into formal proofs in the language of a proof assistant. Large language models are also used to assist mathematicians in such proofs [59][60].

While proving theorems is an essential part of mathematical activity, theorems are perhaps better seen as the fruit of deeper labor: good definitions, which capture the phenomenon in question, simplify the network of theorems, and shorten proofs. Even further upstream from formal definitions are the *technical insights* that experts synthesize, which are often articulable in more than one formal way.

This is the natural next step for conventional "autoformalization." Alex Altair has proposed a trajectory of math skill which AI should catch up to quickly (see Figure 3) [61]:

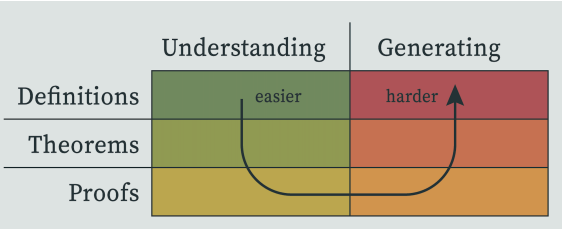



*Figure 3.* Trajectory of mathematical skill development, showing progression from formal to post-rigorous stages [61].

In addition to AI-assisted proofs, we contend that AI-assisted *definitions* (created from human discussions) may allow meta-formal knowledge to scale directly. In his blog [62], Terence Tao notes that mathematical skill does not culminate in formalisms, but extends beyond into a "post-rigorous stage" characterized by intuitive manipulation that can be "converted into a rigorous argument whenever required."

Mathematicians, he says,

> *"no longer need the formalism in order to perform high-level mathematical reasoning, and are actually proceeding largely through intuition, which is then translated (possibly incorrectly) into formal mathematical language."*

While this phenomenon is easy enough to experience first-hand within a subfield, **it does not *scale*.** Despite being truly mathematical activity (as Tao claims) and possessing real technical content, the "intuitive" nature of post-rigorous reasoning means it is not yet granted first-class citizenship as a mathematical language or object.

**Example: 'Inflation as conflict' as a meta-formalism**

In their paper *Inflation is Conflict* [63], Lorenzoni and Werning explore a technical intuition that the proximal cause of inflation is conflict over relative prices. Importantly, instead of presenting one unified model, they present a single intuitive *insight* and compute its implications in multiple *different* formal models. They conclude with the remark (emphasis ours):

> *In our view, traditional ideas and models of inflation have been very useful, but are either incomplete about the mechanism or unnecessarily special. The broad phenomena of inflation deserves a **wider and more adaptable framework**, much in the same way as growth accounting is useful and transcends particular models of growth. The conflict view offers exactly this, a framework and concept that **sits on top of** most models. Specific fully specified models can provide different stories for the root causes, as opposed to proximate causes, of inflation.*

Much like the concept of inflation, we expect many technical concepts to resist a single formalization. In particular, fundamental notions in AI safety that have resisted definitional consensus—such as "deception," "harm," "power-seeking," "autonomy"—could all similarly "deserve a wider and more adaptable framework" that "sits on top of" (and are translated into) specific formal definitions.

### 4.3. Framing

The design considerations for live theory begin with the assumption of a "middle period" [64] of mildly intelligent AI that operates at extremely low-cost and low-latency, such that intelligence becomes an infrastructure, backgrounded in the same way as money, electricity and the Internet have.

Amongst the socio-technical changes associated with this period, we posit that every researcher will have access to capable AI "math agents." These are able to produce near-instant valid mathematics from informal natural language prompts. We assume the following:

1. **AI math agents are more than just proof copilots:** We assume that math agents not only formalize natural language mathematical statements and assist with proofs and lemmas, but also assist with creating definitions and models (and a subsequent body of relatively original mathematics) from informal suggestions.
2. **AI math agents are not super-intelligent:** Although they are able to "autocomplete" mathematical suggestions, they remain unable to autonomously attune to relevance and *taste*, much like language models of today. They are moderately creative, but still need guidance

as a graduate student might, both in the production of a mathematical body of work and the consumption (i.e., application) of it.

These predictions are not meant to hold for the indefinite future, but only a middle period where we might reframe the entirety of the alignment problem and how to approach it. In this way, we in fact leave addressing the alignment problem to the near future, only equipping it with a new ontology for theorization.

Under these assumptions, the key insight supplied by live theory is to alter the way we generalize, shifting the focus from formal artefacts to post-formal ones.

In the current paradigm, we generalize via the operations of *abstraction*, then *instantiation*. We first *abstract* observations and their invariant pattern into a conceptual core (a formalism that "captures" the pattern in a context-free way). Then, to apply the insights, we *instantiate* the rigid parametric structure (variables) with contextual data that can fit the abstracted pattern in a strict, formulaic way (i.e., with values that match the variable type).

With live theory, we shift from formal artefacts ("theories") to more informal "theory-prompts." These can be "rendered" using moderately intelligent math agents into relevant formalisms according to the application context.

These post-formal artefacts, unlike a traditional theory or formalism, would

- capture a concept in a *family* of formalisms that cannot be parametrically related to one another;
- represent a mixture of formal *and informal* information about the concept; a "theory-prompt" created by experts who have theoretical insights, translated into formalisms by AI as needed.

However, these artefacts would also, like a traditional theory or result,

- be a *portable* artefact that can be exchanged, iterated, and played around with;
- be *applied* in combination with an "application-prompt" that captures application-relevant information, created by the applied practitioners in a domain.

In commodifying these inputs (i.e., the postformal "theory-prompts"), we make them easy to transfer, iterate, and collaborate on, much like traditional mathematical artefacts. We posit them as the new locus of human research activity.

We've presented the just-in-time formalism generation process in Figure 4 and a basic infrastructure diagram of producers, consumers, and AI math assistants in Appendix A.

### 4.4. In practice: Prototype and MoSSAIC

We've frequently cited examples where a non-formulaic responsivity is required in order to, e.g., tailor algorithms to run well on GPUs.

We believe that AI will be able to perform such responsive tailoring of insights to substrates, and this has both negative and positive ramifications. In Figure 5, we re-display the nested contexts given in Figure 1, but indicate the increasing domain of autonomous reconfiguration.

Producing and engineering this nesting is something that can only be performed by attending to the specific details of the substrate. This activity is creative, responsive, active tailoring; it does not scale, hence the development of academic subfields of human researchers finding solutions that fit the substrates they work with (see Figure 6).

Our threat model is based on the fact that advanced intelligence will be able to apply similar creativity in its search for ways to evade our attempts to interpret and control it. Our *opportunity model* is that we might leverage this same responsivity in the tools we use to *understand and mitigate* these risks.

We anticipate that moderately intelligent math agents that can support the transfer of post-formal insights to novel substrates will mean that tasks requiring many hours of specialized human labor today will become as simple and quick as changing a variable.

This is what we mean by "keeping pace with intelligence." To track and counter the substrate-flexibility of our threat model, we design a similar (i.e., at or exceeding the pace of) substrate-flexible solution for tasks upstream of risk-management.

In other words, we have more substrate flexibility in our conceptualization and interventions. These should be deployable at least as early (and preferably much earlier) as the deployment of agents that increase the substrate-flexibility of the threat.

To make this speculative proposal easier to engage with, we have presented some of the prototypes we are currently developing to implement the abovementioned research infrastructure, in Appendix B.

### 4.5. What MoSSAIC is not

We should stress a few things that we view live theory as separate from. Firstly, it is not about "autonomous AI scientists." The use of AI is backgrounded as a tool to assist with the communication of intuitions, ultimately based on intuitions and insights that come from humans.

We believe that the core steering role humans play makes this theory of change more subtle than the idea of simply

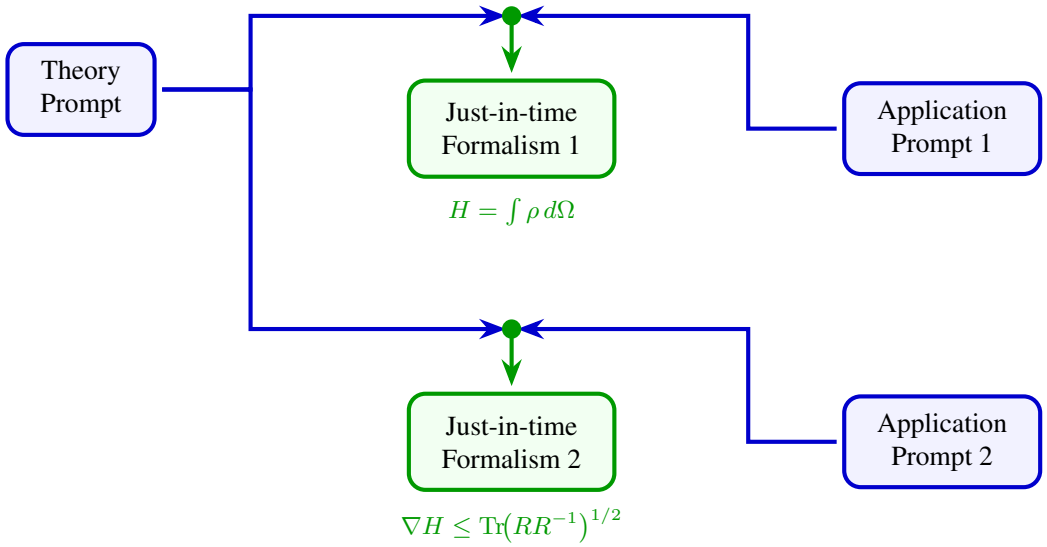



*Figure 4.* A theory-prompt is translated into a just-in-time formalism for each context, informed by the local application prompt

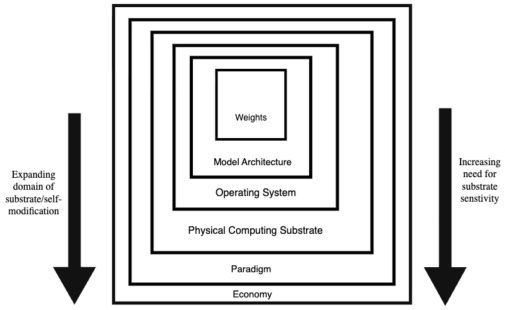



*Figure 5.* As agency increases, the domain of autonomous reconfiguration expands into increasingly wide contexts, requiring greater substrate-sensitivity.

"using AI to solve AI safety." Instead, MoSSAIC looks to develop frameworks, tools, and intelligent infrastructure for porting human insights between different contexts, which we claim is the truer desideratum underlying "general" methods (or oversight that "composes" or "scales"). We will still as researchers need to lay out seed ideas (such as "deception" or "power-seeking") and guide their development.

This proposal contains many aspects that remain speculative. However, we argue that thinking carefully about the *opportunity model* is essential to meeting the threat model.

To say more about what sets this proposal apart from just "use AI to align AI", our emphasis is on the moderate creativity of medium-term intelligence, and how to leverage that. More specifically, in gearing towards a live-theoretic infrastructure, we aim to supply a sociotechnical [65] ontology for subsequent reframings and development of the ongoing task of alignment,[14] now freed from a notion of progress that is tied to formulaic abstractions and practices alone. Instead of a generic proposal, we're providing specifics of the setup as noted above.

We also argue that if you do anticipate radical transformation from AI, you should anticipate moderately radical change in the medium term, however small the interim. This interim period may be quite short, and yet the amount of cognitive effort that is appropriate to devote to the design could be extremely large, given the potential impact that follows from it.

## 5. Author Contributions

Sahil K seeded the initial ideas and supervised the project. Matt Farr developed the arguments at MATS 6.0 and co-wrote the initial and current drafts of the paper with Chris Pang and Aditya Arpitha Prasad, respectively. Chris developed many of the examples and connections to computer science, Aditya helped shape the ML aspects of the paper.

[14] Informally, we intend to treat "alignment" as a verb rather than a noun

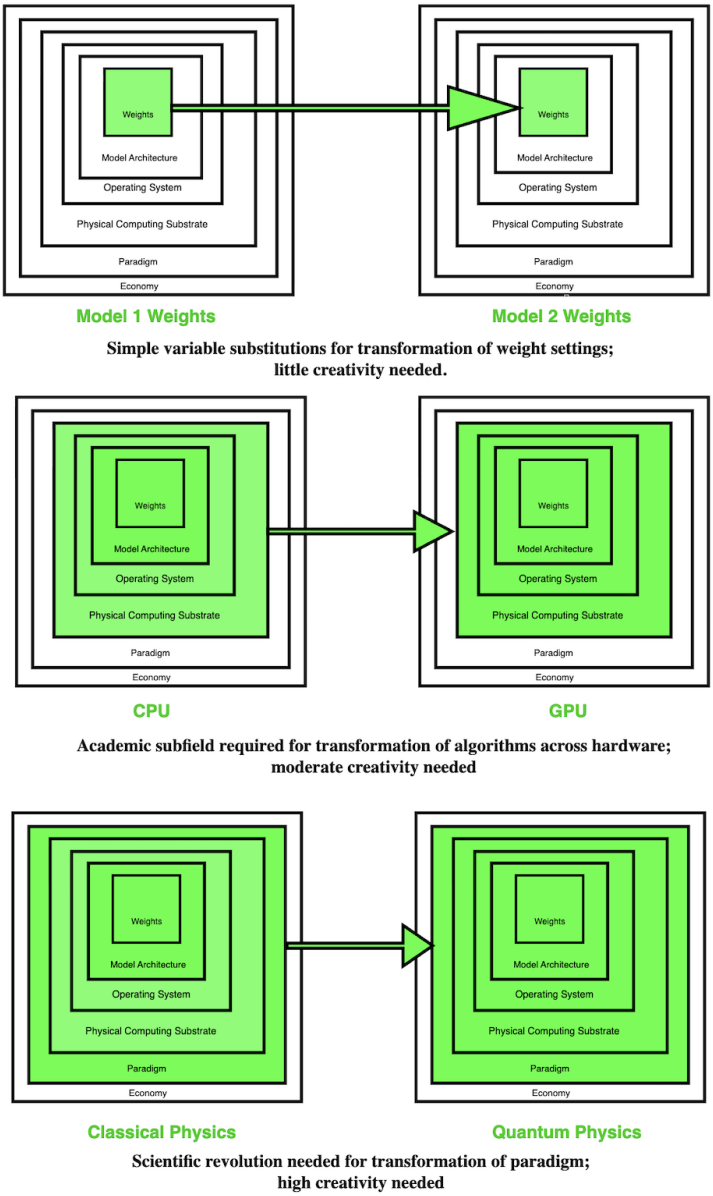



*Figure 6.* The broader the context/substrate change, the more creativity and time needed for transformation. Since these changes also pose more complex risks, risk mitigation needs to scale at pace with increasing capability and agency.

## References


[1] Kola Ayonrinde and Louis Jaburi. A mathematical philosophy of explanations in mechanistic interpretability – the strange science part i.i, 2025. URL https://arxiv.org/abs/2505.00808.

[2] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2025. URL https://arxiv.org/abs/2310.01405.

[3] Leonard Bereska and Efstratios Gavves. Mechanistic interpretability for ai safety – a review, 2024. URL https://arxiv.org/abs/2404.14082.

[4] Dan Hendrycks, Mantas Mazeika, and Thomas Woodside. An overview of catastrophic ai risks, 2023. URL https://arxiv.org/abs/2306.12001.

[5] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits, 2020. URL https://distill.pub/2020/circuits/zoom-in/. Accessed on 25/6/2025.

[6] Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, Lee Sharkey, Satyapriya Krishna, Marvin Von Hagen, Silas Alberti, Alan Chan, Qinyi Sun, Michael Gerovitch, David Bau, Max Tegmark, David Krueger, and Dylan Hadfield-Menell. Black-box access is insufficient for rigorous ai audits. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 2254–2272. ACM, June 2024. doi: 10.1145/3630106.3659037. URL http://dx.doi.org/10.1145/3630106.3659037.

[7] Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 2020. doi: 10.1162/tacl_a_00321. URL https://aclanthology.org/2020.tacl-1.25/.

[8] Giuseppe Casalicchio, Christoph Molnar, and Bernd Bischl. *Visualizing the Feature Importance for Black Box Models*, page 655–670. Springer International Publishing, 2019. ISBN 9783030109257. doi: 10.1007/978-3-030-10925-7_40. URL http://dx.doi.org/10.1007/978-3-030-10925-7_40.

[9] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017. URL https://arxiv.org/abs/1703.01365.

[10] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2): 336–359, October 2019. ISSN 1573-1405. doi: 10.1007/s11263-019-01228-7. URL http://dx.doi.org/10.1007/s11263-019-01228-7.

[11] Daniel Tan, David Chanin, Aengus Lynch, Dimitrios Kanoulas, Brooks Paige, Adria Garriga-Alonso, and Robert Kirk. Analyzing the generalization and reliability of steering vectors, 2025. URL https://arxiv.org/abs/2407.12404.

[12] Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations, 2023. URL https://arxiv.org/abs/2307.15771.

[13] Open Philanthropy. Request for proposals: Technical ai safety research, 2025. URL https://www.openphilanthropy.org/request-for-proposals-technical-ai-safety-research/. Accessed on 12/05/2025.

[14] Aaron Mueller, Jannik Brinkmann, Millicent Li, Samuel Marks, Koyena Pal, Nikhil Prakash, Can Rager, Aruna Sankaranarayanan, Arnab Sen Sharma, Jiuding Sun, Eric Todd, David Bau, and Yonatan Belinkov. The quest for the right mediator: A history, survey, and theoretical grounding of causal interpretability, 2024. URL https://arxiv.org/abs/2408.01416.

[15] Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Bloom, Stella Biderman, Adria Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, Eric J. Michaud, Stephen Casper, Max Tegmark, William Saunders, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Tom McGrath. Open problems in mechanistic interpretability, 2025. URL https://arxiv.org/abs/2501.16496.

[16] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition, 2022. URL https://arxiv.org/abs/2209.10652.

[17] Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability, 2025. URL https://arxiv.org/abs/2301.04709.

[18] Joshua Engels, Eric J. Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear, 2025. URL https://arxiv.org/abs/2405.14860.

[19] Sid Black, Lee Sharkey, Leo Grinsztajn, Eric Winsor, Dan Braun, Jacob Merizian, Kip Parker, Carlos Ramón Guevara, Beren Millidge, Gabriel Alfour, and Connor Leahy. Interpreting neural networks through the polytope lens, 2022. URL https://arxiv.org/abs/2211.12312.

[20] Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets, 2024. URL https://arxiv.org/abs/2310.06824.

[21] Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens, 2023. URL https://arxiv.org/abs/2303.08112.

[22] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models, 2023. URL https://arxiv.org/abs/2309.08600.

[23] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

[24] Luke Bailey, Alex Serrano, Abhay Sheshadri, Mikhail Seleznyov, Jordan Taylor, Erik Jenner, Jacob Hilton, Stephen Casper, Carlos Guestrin, and Scott Emmons. Obfuscated activations bypass llm latent-space defenses, 2025. URL https://arxiv.org/abs/2412.09565.

[25] Stewart Brand. *The Clock of the Long Now: Time and Responsibility*. Basic Books, New York, 1999. ISBN 046504512X.

[26] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., New York, NY, USA, 1982. ISBN 0716715678.

[27] Paul Rendell. This is a universal turing machine (utm) implemented in conway's game of life., 2010. URL http://rendell-attic.org/gol/utm/index.htm. Accessed on 12/10/2024.

[28] Intel Corporation. Intel® core™ processor family, 2024. URL https://www.intel.com/content/www/us/en/products/details/processors/core.html. Accessed on 12/12/2024.

[29] Scott Aaronson. Quantum computing: Between hope and hype, 2024. URL https://scottaaronson.blog/?p=8329. Accessed on 08/15/2025.

[30] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[31] Sara Hooker. The hardware lottery, 2020. URL https://arxiv.org/abs/2009.06489.

[32] Anirudh Srikanth, Carlotta Trigila, and Emilie Roncali. GPU optimization techniques to accelerate optiGAN—a particle simulation GAN. *Machine Learning: Science and Technology*, 5(2):027001, June 2024. doi: 10.1088/2632-2153/ad51c9.

[33] Barry Schluntz, Erik; Zhang. Building effective agents, 2024. URL https://www.anthropic.com/research/building-effective-agents. Accessed on 12/19/2024.

[34] Google DeepMind. Project astra, 2024. URL https://deepmind.google/technologies/project-astra/. Accessed on 12/19/2024.

[35] Anthropic. Build with claude: Computer use (beta), 2024. URL https://docs.anthropic.com/en/docs/build-with-claude/computer-use. Accessed on 12/19/2024.

[36] Marianna Nezhurina, Lucia Cipolina-Kun, Mehdi Cherti, and Jenia Jitsev. Alice in wonderland: Simple tasks showing complete reasoning breakdown in

state-of-the-art large language models, 2024. URL https://arxiv.org/abs/2406.02061.

[37] Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability, 2023. URL https://arxiv.org/abs/2304.14997.

[38] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[39] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

[40] Ameen Ali, Itamar Zimerman, and Lior Wolf. The hidden attention of mamba models, 2024. URL https://arxiv.org/abs/2403.01590.

[41] Holden Karnofsky. Forecasting transformative ai, part 1: What kind of ai?, 2021. URL https://www.cold-takes.com/transformative-ai-timelines-part-1-of-4-what-kind-of-ai/. Accessed on 12/13/2024.

[42] Nick Bostrom. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, Inc., USA, 1st edition, 2014. ISBN 0199678111.

[43] Roman V Yampolskiy. From seed ai to technological singularity via recursively self-improving software. *arXiv preprint arXiv:1502.06512*, 2015.

[44] OpenAI. Introducing openai o3 and o4-mini, 2025. URL https://openai.com/index/introducing-o3-and-o4-mini/. Accessed on 12/19/2024.

[45] Adrian Thompson. An evolved circuit, intrinsic in silicon, entwined with physics. *Lecture Notes in Computer Science*, 1259, 01 1997. doi: 10.1007/3-540-63173-9_61.

[46] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models, 2023. URL https://arxiv.org/abs/2211.15661.

[47] Nate Soares. Deep deceptiveness, 2023. URL https://www.alignmentforum.org/posts/XWwvwytieLtEWaFJX/deep-deceptiveness. Accessed on 12/19/2024.

[48] Andrew Critch. What multipolar failure looks like, and robust agent-agnostic processes (raaps), 2021. URL https://www.lesswrong.com/posts/LpM3EAakwYdS6aRKf/what-multipolar-failure-looks-like-and-robust-agent-agnostic. Accessed on 12/4/2024.

[49] Jan Kulveit. Box inversion hypothesis, 2020. URL https://www.alignmentforum.org/posts/TQwXPHfyyQwr22NMh/box-inversion-hypothesis.

[50] Erik Jones, Anca Dragan, and Jacob Steinhardt. Adversaries can misuse combinations of safe models, 2024. URL https://arxiv.org/abs/2406.14595.

[51] Dan Milmo. Sacking, revolt, return: how crisis at openai over sam altman unfolded, 2023. URL https://www.theguardian.com/technology/2023/nov/25/how-crisis-openai-sam-altman-unfolded. Accessed on 09/18/2025.

[52] Rohin Shah. [an #95]: A framework for thinking about how to make ai go well, 2020. URL https://www.lesswrong.com/posts/9et86yPRk6RinJNt3/an-95-a-framework-for-thinking-about-how-to-make-ai-go-well. Accessed on 12/16/2024.

[53] Eliezer Yudkowsky and Nate Soares. Functional decision theory: A new theory of instrumental rationality, 2018. URL https://arxiv.org/abs/1710.05060.

[54] Paul Christiano. My research methodology, 2021. URL https://ai-alignment.com/my-research-methodology-b94f2751cb2c. Accessed on 12/20/2024.

[55] Ajeya; Christiano, Paul; Cotra and Mark Xu. Eliciting latent knowledge: How to tell if your eyes deceive you, 2021. URL https://docs.google.com/document/d/1WwsnJQstPq91_Yh-Ch2XRL8H_EpsnjrC1dwZXR37PC8/edit. Accessed on 12/16/2024.

[56] Unknown. Big o notation tutorial - a guide to big o analysis, 2025. URL https://www.geeksforgeeks.org/dsa/

analysis-algorithms-big-o-analysis/. Accessed on 08/16/2025.

[57] Unknown. Why quicksort is better than mergesort ?, 2022. URL https://www.geeksforgeeks.org/dsa/quicksort-better-mergesort/. Accessed on 08/16/2025.

[58] Pieter Hijma, Stijn Heldens, Alessio Sclocco, Ben van Werkhoven, and Henri E. Bal. Optimization techniques for gpu programming. *ACM Comput. Surv.*, 55(11), March 2023. ISSN 0360-0300. doi: 10.1145/3570638. URL https://doi.org/10.1145/3570638.

[59] Haiming Wang, Mert Unsal, Xiaohan Lin, Mantas Baksys, Junqi Liu, Marco Dos Santos, Flood Sung, Marina Vinyes, Zhenzhe Ying, Zekai Zhu, Jianqiao Lu, Hugues de Saxcé, Bolton Bailey, Chendong Song, Chenjun Xiao, Dehao Zhang, Ebony Zhang, Frederick Pu, Han Zhu, Jiawei Liu, Jonas Bayer, Julien Michel, Longhui Yu, Léo Dreyfus-Schmidt, Lewis Tunstall, Luigi Pagani, Moreira Machado, Pauline Bourigault, Ran Wang, Stanislas Polu, Thibaut Barroyer, Wen-Ding Li, Yazhe Niu, Yann Fleureau, Yangyang Hu, Zhouliang Yu, Zihan Wang, Zhilin Yang, Zhengying Liu, and Jia Li. Kimina-prover preview: Towards large formal reasoning models with reinforcement learning, 2025. URL https://arxiv.org/abs/2504.11354.

[60] Z. Z. Ren, Zhihong Shao, Junxiao Song, Huajian Xin, Haocheng Wang, Wanjia Zhao, Liyue Zhang, Zhe Fu, Qihao Zhu, Dejian Yang, Z. F. Wu, Zhibin Gou, Shirong Ma, Hongxuan Tang, Yuxuan Liu, Wenjun Gao, Daya Guo, and Chong Ruan. Deepseek-prover-v2: Advancing formal mathematical reasoning via reinforcement learning for subgoal decomposition, 2025. URL https://arxiv.org/abs/2504.21801.

[61] Alex Altair. A simple model of math skill, 2024. URL https://www.lesswrong.com/posts/EF8tvShQJ5cbdZzTb/a-simple-model-of-math-skill. Accessed on 29/06/2025.

[62] Terrence Tao. There's more to mathematics than rigour and proofs, 2022. URL https://terrytao.wordpress.com/career-advice/theres-more-to-mathematics-than-rigour-and-proofs/. Accessed on 29/06/2025.

[63] Guido Lorenzoni and Iván Werning. Inflation is conflict. Working Paper 31099, National Bureau of Economic Research, April 2023. URL http://www.nber.org/papers/w31099.

[64] Sahil K. Live theory part 0: Taking intelligence seriously, 2024. URL https://www.alignmentforum.org/posts/QvnzEHvodmwfBXu94/live-theory-part-0-taking-intelligence-seriously. Accessed on 06/29/2025.

[65] Andrew Critch. Safety isn't safety without a social model (or: dispelling the myth of per se technical safety), 2024. URL https://www.alignmentforum.org/posts/F2voF4pr3BfejJawL/safety-isn-t-safety-without-a-social-model-or-dispelling-the. Accessed on 08/22/2025.

# Appendix

Appendix B - Diagram

## A. Live Theory Infrastructure Diagram

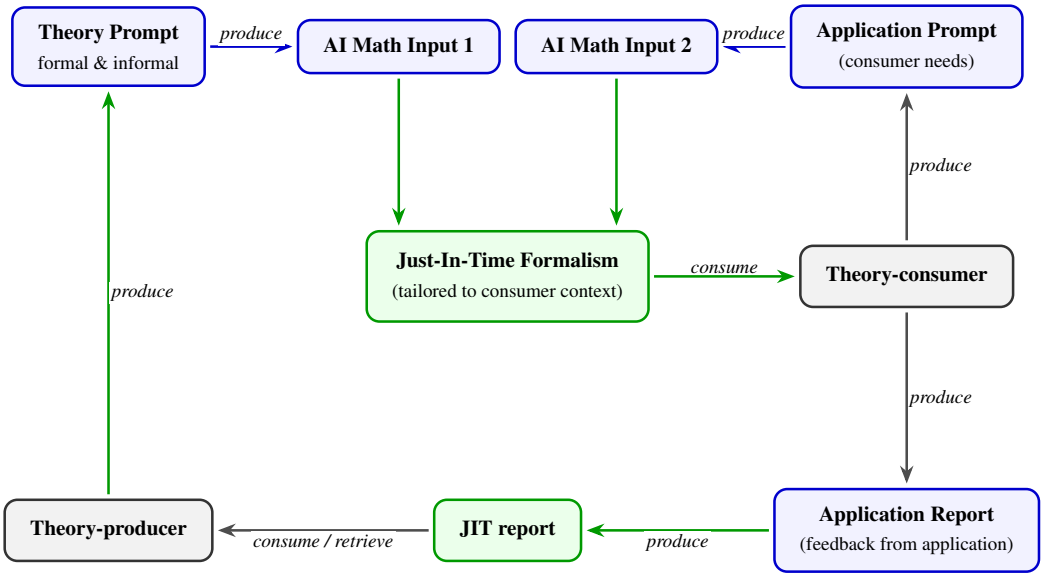



*Figure 7.* The AI math agent takes in two inputs, one from the theory prompt and another from the application prompt to create a just-in-time formalism. The consumer also produces an application report that captures the empirical utility of the formalism; many such reports are retrieved by a human theory-producer to refine their theory-prompts

## B. Research Tools Based on Live Theory

### B.1. Research Support in the Age of AI

We present some initial design prototypes we are currently building around the live theory framework. In particular, we want to demonstrate the kinds of flexible, contextual understanding and feedback processes that facilitate substrate-flexibility, as described in Section 4.

These research tools are based around the following two claims:

- **Claim 1:** Current trends suggest we are heading towards AI models with lower latencies, lower costs, and greater adaptivity. Hence, we should build research tools to more fully exploit the selective advantages that AI will offer over the next few years.

- **Claim 2:** There are some things that will still require human input. In the context of AI safety, we posit that humans are much better able to identify and isolate subtle connections between phenomena, even when those phenomena lack a unified or formal description.

Our aim is to use the former to allow us to scale the latter. We are developing tools that offload some of the cognitive work to AI, whilst minimizing disruption to the human processes of insight generation. We are designing sensitive research tools that *passively add rigor to human conversations*, to exploit the productive tension between informal and formal outputs.

### B.2. Sensitivity in Research Conversation

To demonstrate how we might carry out research with the help of sensitive AI-powered tools, we are designing the following pipeline:

1. **Insight Extraction:** As conversations happen, insights can be continually marked and extracted into a directed acyclic graph (DAG) format. This structure can then be explored both in the time and context dimensions. This tool is called Live Conversational Threads (LCT) and it performs this extraction from real time audio.

2. **Formalism Generation:** In keeping with live theory's orientation towards postformal artefacts, we are working on a tool that integrates with LCT. When prompted with a user's context, it generates formalisms from the gathered insights.

3. **Discernment of Outputs:** To prevent an influx of mathematically sound yet vacuous formalisms, we are developing an interface that enhances human capacity to discern the relevance of the generated mathematical artefacts.

### B.3. Part 1: Live Conversational Threads (LCT)

We want to capture insights from researcher interactions, minimizing the disruption involved in noting these down and tracking how the conversation develops. LCT is a tool that allows users to capture potential insights from conversations.

#### B.3.1. Core Functions

LCT[15] captures "threads" (independent parts of a conversation) and the thematic flow of context between them using a DAG structure. Navigating the nodes of this graph allows you to follow the flow of dialogue more naturally.

Figure 8 showcases this tool in action.

[15] You can try our demo here - https://lct-app-515466416372.us-central1.run.app/

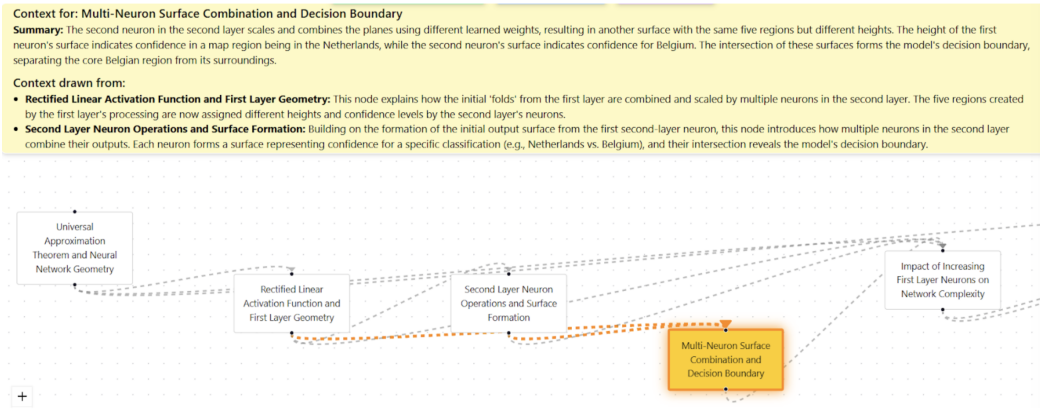



*Figure 8.* DAG structure showing conversational threads and their thematic connections in LCT. Note how the raw transcript, summary, and details on which notes are related are all visible upon clicking a node.

As seen below in Figure 9, this tool allows users to mark points in a conversation where they intuitively identify contextual progress or sense a potential insight.

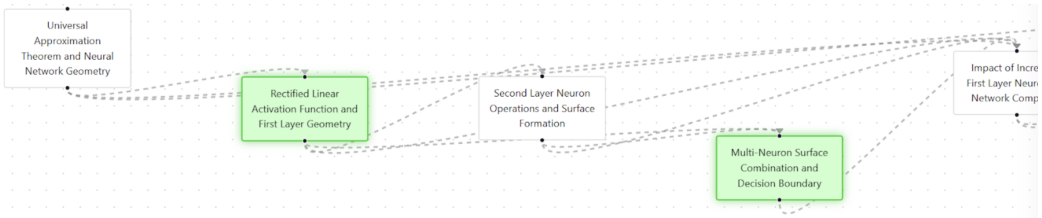



*Figure 9.* Marked insights and contextual progress points in conversations are highlighted

### B.4. Part 2: Formalism Generation

In addition to marking potential progress in a conversation, we want to be able to render any potential insights into a portable format. We plan to implement a further formalism generation tool that exploits AI capabilities to autoformalize natural language statements into mathematically valid formulas.

The consumers of these conversational insights can provide information about their specific research interests or the local context of the substrate in which they are working (see Figure 10).

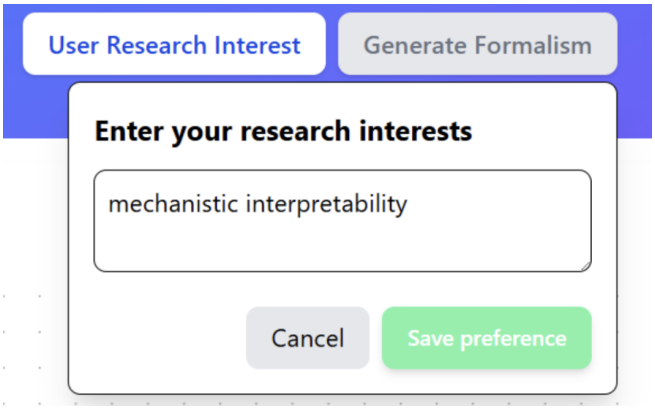



*Figure 10.* Context information can be inputted and leveraged for formalism generation.

The AI-enabled infrastructure operates seamlessly in the background considering the potential insight and local context to produce personalized substrate-sensitive formalisms (Figure 11).

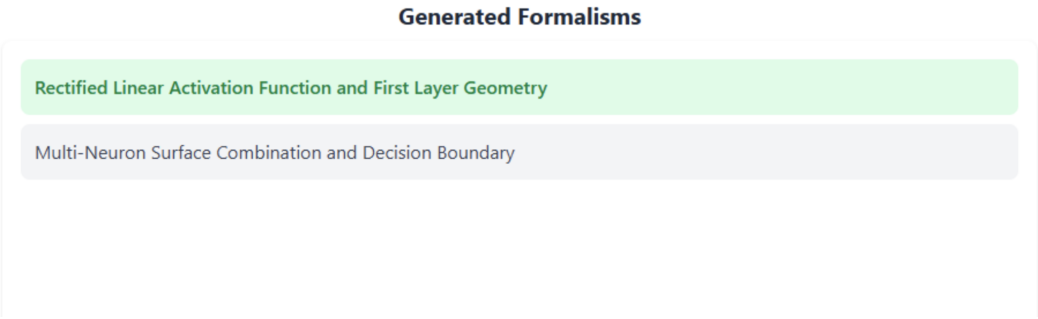



*Figure 11.* List interface showing conversational threads available for formalism generation and the retrieval of relevant conversations

Initial formalisms are modelled using causal loop diagrams (see Figure 12). This structure is simple enough to allow for quick modifications and yet rich enough to capture a lot of the underlying dynamics we care about.

*Figure 12.* Example causal loop diagram generated from conversational thread analysis.

This can further be made rigorous by using the toggle button. We can interoperate this causal loop diagram into a provably valid mathematical formula (as shown in Figure 13) that can be represented in languages such as Lean to verify correctness.

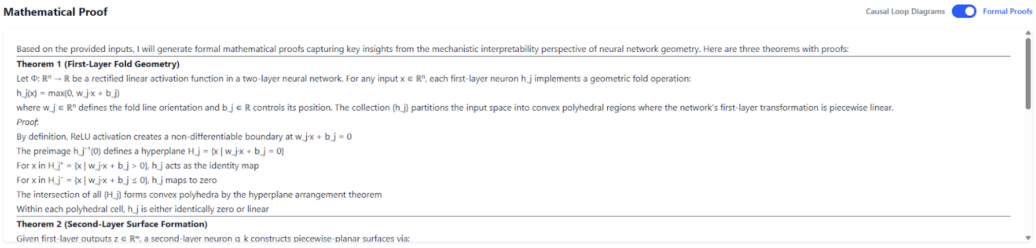



*Figure 13.* Example mathematical proof generated by DeepSeek Prover v2 from conversational insights.

### B.5. Part 3: Live Discernment

This system can generate perfectly valid mathematical constructions. However, it takes skill to interpret such formalisms and ensure their relevance.[16] To avoid an explosion of trivial AI-generated formalisms, we are designing tools to augment

[16]There is a risk that malicious actors might take advantage of ("It's produced by AI, so it's correct") to support fraudulent claims. There is documented evidence of this in scientific publications, and we posit this will get worse with the advance of these math-capable systems.

human powers of discerning relevance, as we are convinced that these advanced mathematical AIs can generate correct/valid formalisms.

#### B.5.1. When Discernment is Relevance-Sensitive

A formalism (model) that is relevant to one "domain" does not mean that the relevance will automatically transfer to another. One has to distill the insight of the formalisms to an invariant that can be ported to another domain, and after porting, the relevance of the insight can be actuated in the language of the target domain. Our interface allows users to engage with formalisms across various domains and at multiple levels of granularity.

For instance, one can adopt many "lenses" with an economic paper. On the "surface level," one can look through the economic-theory lens, or one can zoom in to look at the statistical methods and conclusions, or even further to explore the mathematical model in more detail. Each lens maintains connections to the queries the user is using to investigate the paper.

We are also developing collaborative aspects of the interface.

There will be as many "applications" of insightful formalisms as there are "unique relevance perspectives". This happens when we have the tools to "identify" in a sensitive way when a particular insight/composition of insights with a "peer-defined" relevance prompt. Such a tool would take an insight (this could be the insight prompt of a particular formalism of it) and a relevance, and apply discernment (this could be human curated or suggestions from an AI) on how the insight fits the "problem" in a sensitive way. Users can then navigate the discernment space by either zooming in on particular details of the formalisms or the problem at hand and adding queries. They can also zoom out and look at the entire inference pipeline of the particular formalism while viewing connections that are relevant to the problem at hand.

A more advanced version of live-discernment would look something like this: Once a user makes contact with the Live-discernment system, they can add the formalisms of interest (the artifacts they would like to discern) to the system. Further, they can add their expressions of relevance to the said instance of the system and, from here, they can start exploring the formalisms via the expressions they have included. This will open up a "unique inference pipeline" that represents the path(s) of discernment the user took. These paths are linked to the particular formalism, and if any other peer is looking at the formalisms, they have the option to explore other peers' pipelines and incorporate them into their own.