

Behavior Theory: The Static Collapse Theorem

Heath Fortin

May 29, 2025

1 Definitions

- **Problem Universe:** $\mathcal{P} = \{P_1, P_2, \dots\}$.
- **Solution Space:** For each $P \in \mathcal{P}$, the valid outputs lie in $\mathcal{S}_P \subseteq \mathcal{O}$.
- **Agent:** A system with finite memory, $|\omega| \leq K < \infty$.
- **Static System:** Parameters ω frozen after deployment.

2 Core Axioms

- A1. Task-Oblivious Execution.** The agent processes a fixed input x_* with no privileged task-identifying information.
- A2. Monolithic Output Requirement.** The agent must emit a single output $y \in \mathcal{O}$ satisfying *all* tasks:

$$y \in \bigcap_{i=1}^n \mathcal{S}_{P_i}.$$

- A3. Independent Solution Sets.** There exists a measure μ on \mathcal{O} and a constant $\rho < 1$ such that for each i ,

$$\mu(\mathcal{S}_{P_i}) \leq \rho,$$

and the \mathcal{S}_{P_i} are μ -independent, giving

$$\mu\left(\bigcap_{i=1}^n \mathcal{S}_{P_i}\right) \leq \prod_{i=1}^n \mu(\mathcal{S}_{P_i}) \leq \rho^n.$$

Note: This applies to solution *sets* \mathcal{S}_{P_i} , not individual outputs y .

3 Interpretation Notes

- **Task Information:** Even if x contains task labels, fixed parameters ω prevent any runtime reconfiguration.
- **Monolithic Output:** The requirement $y \in \bigcap_i \mathcal{S}_{P_i}$ illustrates the limitations of static capacity.
- **Worst-Case Assumption:** A3 models an exponential “collapse.” Overlap between sets may slow it down but cannot stop it.

4 Main Theorem

Theorem 4.1 (Static Collapse). *Under A1–A3, any static agent facing n tasks satisfies*

$$\Pr\left[y \in \bigcap_{i=1}^n \mathcal{S}_{P_i}\right] \leq \rho^n,$$

and across the entire class of capacity- K agents,

$$\Pr[\exists \text{ successful agent}] \leq 2^K \rho^n \xrightarrow{n \rightarrow \infty} 0.$$

Proof. The single-agent bound is immediate from A3. Since there are at most 2^K distinct settings of ω , a union bound over all agents gives the class-wide result. \square

5 Recursive Collapse Corollary

Corollary 5.1. *Let $h: \mathcal{X} \rightarrow \mathcal{P}$ be a static task-classifier with $|\omega_h| \leq K_h$, and let $\mathcal{T}_k \subseteq \mathcal{P}$ be valid ID sets. Then*

$$\Pr\left[h(x_*) \in \bigcap_{k=1}^n \mathcal{T}_k\right] \leq 2^{K_h} \tau^n \xrightarrow{n \rightarrow \infty} 0.$$

No static subsystem escapes collapse.

6 Broader Implications

Static systems cannot generalize indefinitely across unbounded tasks. Even optimistic overlap does not prevent eventual performance degradation without capacity growth. True generalization requires systems whose internal structure or memory can change post-deployment.

7 Dynamic Intelligence Principle

- **Escape Hatch:** Only by increasing $|\omega_t|$ at runtime—e.g. via external memory, module synthesis, or on-the-fly architecture changes—can an agent avoid collapse.
- **Collapse Test:**

If $|\omega_t| \uparrow \Rightarrow \text{Dynamic} \Rightarrow \text{No collapse}; \quad \text{else Static} \Rightarrow \text{Collapse}.$

Edit Notes

1. Independence of solution sets avoids the trivial case where tasks are identical.
2. Any agent not continuously learning faces an ever-increasing set of indistinguishable solutions.
3. The task-oblivious assumption parallels Gödel’s incompleteness: without growth, any static system is bound to “miss” truths.