Monitor-Generate-Verify (MGV): **Formalising Metacognitive Theory for Language Model Reasoning**

Nick Oh
socius labs
London, UK
nick.sh.oh@socius.org

Fernand Gobet
Centre for Philosophy of
Natural and Social Science (CPNSS)
London School of Economics
London, UK
f.gobet@lse.ac.uk

Abstract

Test-time reasoning architectures such as those following the *Generate–Verify* paradigm – where a model iteratively refines or verifies its own generated outputs – prioritise generation and verification, but exclude the monitoring processes that determine when and how reasoning should begin. This omission may contribute to the prefix dominance trap, in which models commit early to suboptimal reasoning paths and seldom recover, yielding roughly 20% accuracy loss. We address this architectural gap by formalising Flavell's and Nelson and Narens' metacognitive theories into computational specifications, proposing the Monitor–Generate–Verify (MGV) framework. MGV extends the Generate–Verify paradigm by adding explicit monitoring that captures metacognitive experiences (from difficulty assessments to confidence judgements) before generation begins and refines future monitoring through verification feedback. Though we present no empirical validation, this work provides the first systematic computational translation of foundational metacognitive theories, offering a principled vocabulary for understanding reasoning system failures and suggesting specific architectural interventions for future test-time reasoning designs.

1 Introduction

Once language models commit to an initial reasoning strategy, subsequent verification rarely helps; this *prefix dominance trap* causes nearly 20% performance degradation when models choose suboptimal approaches, with virtually no recovery possible through refinement [Luo et al., 2025]. Today's dominant Generate-Verify (G-V) test-time reasoning architectures [Weng et al., 2023, Madaan et al., 2023, Lee et al., 2025, Zhang et al., 2024] exemplify this limitation through their very design. They operate through immediate generation followed by iterative refinement, without assessing task characteristics or selecting appropriate strategies before generating solutions.

Yet, this paradigm omits an implicit phase present in human cognition, where metacognitive monitoring precedes action. Before attempting complex tasks, humans – often without conscious deliberation – assess difficulty, retrieve relevant strategies, and establish confidence criteria [Flavell, 1979, Nelson and Narens, 1990]. This metacognitive capacity operates by transforming world-centred uncertainty into self-centred propositional confidence, enabling both prospective planning and retrospective evaluation through mechanisms including global broadcast and post-decisional evidence accumulation [Fleming, 2024]. Current reasoning architectures – whether Self-Verification [Weng et al., 2023] validating through backward verification of conclusions, SELF-REFINE [Madaan et al., 2023]

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Foundations of Reasoning in Language Models (FoRLM).

improving through iterative feedback and refinement, or ReVISE [Lee et al., 2025] rethinking trajectories through intrinsic self-verification – lack these pre-generation assessment mechanisms, blindly committing to initial approaches that determine ultimate success or failure.

Although the Generate-Verify paradigm successfully mirrors Generation-Recognition models of memory recall [Bahrick, 1970, Kintsch, 1978], capturing how retrieval involves generation followed by recognition-based evaluation, these models specify only the mechanics of recall (object-level mechanics) – lacking the metacognitive control and monitoring that determines *when* and *how* these mechanics should be deployed (meta-level).

Cognitive psychology shows that successful task completion requires precisely such meta-level governance through monitoring task difficulty, selecting strategies, and deciding when to persist or abandon approaches [Flavell, 1979, Nelson and Narens, 1990]. Flavell [1979] demonstrated that cognitive regulation emerges from dynamic interactions between metacognitive knowledge (beliefs about cognitive capabilities, task demands, and strategy effectiveness), metacognitive experiences (feelings of difficulty and comprehension), goals and strategies, with monitoring necessarily preceding action to enable appropriate strategy selection. Nelson and Narens [1990] further specified the hierarchical architecture of metacognition, distinguishing object-level processes from meta-level oversight that operates through control and monitoring flows. Their framework reveals how metacognitive experiences (Feeling of Knowing, Judgements of Learning) guide immediate decisions, with confidence thresholds dynamically adjusting through satisficing to balance accuracy against search costs.

Informed by these cognitive science foundations, we develop algorithmic formalisations that translate psychological theories into computational specifications suitable for reasoning systems. We formalise Flavell's cognitive monitoring model and Nelson and Narens' metamemory frameworks, translating them into computational algorithms to develop `Monitor-Generate-Verify` (`MGV`), a framework where explicit metacognitive monitoring precedes generation. Our formalisations translate abstract psychological constructs into algorithmic structures, highlighting gaps between human cognitive capabilities and current reasoning architectures.

The remainder of this paper is organised as follows: Section 2 situates our contribution within current reasoning architectures and partial metacognitive implementations. Section 3 presents algorithmic formalisations of Flavell's and Nelson and Narens' theories, translating psychological concepts into computational structures – to our knowledge, the first systematic translation of these foundational theories into computational specifications. Section 4 demonstrates how these formalisations explain specific failures and suggest architectural improvements. Section 5 acknowledges implementation challenges, before Section 6 synthesises contributions and future directions.

2 Related Work

2.1 Metacognitive Capabilities in Language Models

Recent research explores metacognitive capabilities in LLMs through behavioural and neural analysis. Behavioural studies examine how models express uncertainty through confidence verbalisation [Wang et al., 2025, Tian et al., 2023, Xiong et al., 2023, Cash et al., 2024, Griot et al., 2025]. Yet implicit confidence measures derived from token likelihoods demonstrate greater metacognitive sensitivity than explicitly prompted confidence statements [Xiong et al., 2023]. This discrepancy between internal representation and external expression indicates that models access metacognitive signals they cannot adequately articulate – Lindsey et al. [2025] demonstrate that while `Claude-3.5-Haiku` accurately reports intermediate computational steps, it fabricates non-existent processes when queried about simple addition, despite correctly activating relevant neural mechanisms.

Direct neural analysis yields more substantive findings. Ji-An et al. [2025] identify a "metacognitive space" of lower dimensionality than models' full neural space, indicating that LLMs monitor compressed representations rather than complete computational states. This observation aligns with established cognitive findings that metacognitive monitoring operates on abstracted representations [Reder, 1987]. Furthermore, metacognitive signals such as confidence correspond to linearly separable directions in representation space [Zou et al., 2023, Liu et al., 2023], suggesting structured organisation of these internal monitoring capabilities.

2.2 Theoretical Foundations Gap

Despite the behavioural and neural evidence demonstrating LLM access to metacognitive signals – confidence representations, capability awareness, compressed monitoring spaces – such capabilities remain architecturally isolated. The distinction between signal detection and systematic control manifests in reasoning failures. Models experiencing the prefix dominance trap fail to revise suboptimal strategies [Luo et al., 2025], indicating that any metacognitive signals about task difficulty or strategy inadequacy remain functionally isolated from generation control.

The absence of systematic control architectures reflects a broader theoretical gap. Contemporary implementations incorporate metacognitive components as discrete modules rather than integrated systems. This fragmentation precludes the coordinated monitoring-control loops that Flavell identifies as necessary for strategy selection, and the hierarchical meta-level governance that Nelson and Narens demonstrate underlies successful cognitive regulation.

However, these psychological specifications lack computational translations. No principled framework exists for implementing such metacognitive architectures in artificial systems. This theoretical gap constrains both our understanding of reasoning failures and our capacity to develop systematic solutions. Without formal mappings from cognitive principles to algorithmic structures, implementations remain ad hoc, adopting surface features of metacognition while omitting the functional relationships that enable effective cognitive control.

Our contribution addresses this gap by providing algorithmic formalisations of established metacognitive theories. These formalisations preserve the functional dependencies identified in cognitive science while specifying implementation requirements for computational systems, thereby establishing theoretical foundations for metacognitive reasoning architectures.

3 Monitor-Generate-Verify (MGV)

Flavell [1979] and Nelson and Narens [1990] developed seminal theories of how metacognition coordinates cognitive processes through monitoring and control loops. These frameworks, though developed for human cognition, offer potential blueprints for computational systems. Flavell's model provides a dynamic architecture where metacognitive knowledge and experience guide strategy selection and verification, while Nelson and Narens' metamemory framework specifies how confidence thresholds and adaptive search mechanisms emerge from hierarchical monitoring and control. By computationally formalising these psychological theories, we establish Monitor-Generate-Verify (MGV) as a theoretical framework for understanding how explicit metacognitive mechanisms could address the architectural limitations of current reasoning systems. The following subsections present detailed formalisations that translate these cognitive science insights into algorithmic structures, revealing both what current architectures lack and how metacognitive principles might be operationalised computationally.

3.1 Flavell's Model of Metacognition

Flavell [1979] conceptualises metacognition as a dynamic control architecture comprising four interacting components: *metacognitive knowledge*, *metacognitive experience*, *goals* (or tasks), and *actions* (or strategies). Rather than operating as independent modules, these components form an integrated system characterised by continuous bidirectional influences, positioning metacognition as a self-regulating system capable of adaptive control over cognitive processes. We present the core computational structure below, with a complete mathematical formalisation provided in Appendix A.

3.1.1 Cognitive Monitoring

The regulation process begins with initialisation, where task $\mathcal{T}$ and goal $\mathcal{G}$ establish the initial state $\mathcal{S}_0 = f(\mathcal{T}, \mathcal{G})$. While Flavell [1979] treats goals and tasks as equivalent, we maintain a computational distinction. $\mathcal{T}$ represents the cognitive enterprise while $\mathcal{G}$ specifies success criteria, enabling clearer analysis of metacognitive processes.

The **monitoring** phase activates metacognitive knowledge differently across cycles. Initial cycles rely solely on task-goal combinations, while subsequent cycles incorporate emerging metacognitive experiences from $\tau - 1$ that trigger additional relevant knowledge. According to Flavell [1979], this

Algorithm 1 Flavell's Metacognitive Regulation

Initialise: $\mathcal{S}_0 \leftarrow f(\mathcal{T}, \mathcal{G})$; $\tau \leftarrow 0$
while $\mathcal{S}_\tau$ = ACTIVE **do**
  // MONITOR: Retrieve knowledge & assess experience
  $\mathcal{MK}_\tau \leftarrow$ **if** $\tau = 0$ **then** retrieve($\mathcal{MK}, \mathcal{T}, \mathcal{G}$)
    else $\mathcal{MK}_{\tau-1} \cup$ retrieve($\mathcal{MK}, \mathcal{ME}_{\tau-1}$)
  $\mathcal{ME}^\tau_{\text{difficulty}} \leftarrow$ feel($\mathcal{T}$, Outcomes$_{\tau-1}$) $\oplus$ assess($\mathcal{T}$, $\mathcal{MK}_\tau$)
  // GENERATE: Select & execute cognitive strategy
  $\mathcal{CS}_\tau \leftarrow$ select($s \in \mathcal{MK}_{\text{Strategy}} \mid \mathcal{ME}^\tau_{\text{difficulty}}, \mathcal{MK}_\tau, \mathcal{T}, \mathcal{G}$)
  $\mathcal{CO}_\tau \leftarrow$ execute($\mathcal{CS}_\tau, \mathcal{T}, \mathcal{G}$)
  // VERIFY: Evaluate progress & update knowledge
  $\mathcal{ME}^\tau_{\text{evaluative}} \leftarrow$ assess($\mathcal{CO}_\tau, \mathcal{MK}_\tau$)
  $\mathcal{MS}_\tau \leftarrow$ select($s \in \mathcal{MK}^{\text{meta}}_{\text{Strategy}} \mid \mathcal{ME}^\tau_{\text{evaluative}}$)
  $\mathcal{MO}_\tau \leftarrow$ execute($\mathcal{MS}_\tau, \mathcal{CO}_\tau, \mathcal{MK}_\tau, \mathcal{G}$)
  $\mathcal{MK} \leftarrow$ update($\mathcal{MK}$, Φ_τ) where $\Phi_\tau = (\mathcal{ME}_\tau, \text{Strategy}_\tau, \text{Outcome}_\tau)$
  $\mathcal{S}_{\tau+1} \leftarrow$ **if** goal_achieved($\mathcal{CO}_\tau, \mathcal{G}$) **then** TERMINATE **else** ACTIVE
  $\tau \leftarrow \tau + 1$
end while

knowledge comprises three categories: *agent variables* ($\mathcal{MK}_{\text{Agent}}$) representing learned self-models of performance patterns and processing preferences; *task variables* ($\mathcal{MK}_{\text{Task}}$) capturing knowledge about cognitive situation assessment including information characteristics and task demands; and *strategy variables* ($\mathcal{MK}_{\text{Strategy}}$) encompassing knowledge about the effectiveness of both cognitive strategies (problem-solving procedures) and metacognitive strategies (monitoring and regulation processes). These categories function as an integrated system where task variables diagnose cognitive demands, strategy variables prescribe responses, and agent variables contextualise both within the agent's capabilities.

The monitoring phase also generates metacognitive experiences of difficulty ($\mathcal{ME}^\tau_{\text{difficulty}}$), which Flavell [1979, p. 909] describes as subjective feelings of complexity, comprehension challenges, or sensing that material exceeds current capabilities. These experiences evolve through iterative assessments, progressing from initial coarse feelings to increasingly nuanced evaluations of specific challenge sources.

During the **generation** phase, metacognitive experiences function as computational signals that require interpretation through metacognitive knowledge to guide strategy selection. The process follows a two-phase pattern. First, $\mathcal{MK}_{\text{Strategy}}$ transforms general difficulty signals into precise diagnostic patterns (e.g., "content uncertainty with unknown terms" or "procedural confusion from missing steps"). Second, these refined patterns activate corresponding cognitive strategies. The selected strategy $\mathcal{CS}_\tau$ is then executed to produce cognitive outcomes $\mathcal{CO}_\tau$, generating feedback that provides both task progress information and context for subsequent monitoring.

The **verification** phase evaluates these outcomes, triggering what Flavell [1979, p. 909] describes as additional metacognitive experiences about performance rather than difficulty. These evaluative experiences ($\mathcal{ME}^\tau_{\text{evaluative}}$) activate metacognitive strategies that assess whether outcomes form a coherent whole, appear plausible and consistent with prior knowledge, and provide an avenue to the goal. The specific metacognitive strategy $\mathcal{MS}_\tau$ selected depends on the nature of the evaluative signal: uncertainty about validity triggers plausibility checking, sensing incompleteness activates coherence assessment, and so forth. Notably, these experiences can add to, delete from, or revise the metacognitive knowledge base through Piagetian mechanisms [Flavell, 1963], with the complete experience tuple Φ_τ updating $\mathcal{MK}$ for future cycles.

3.1.2 Memory and Learning Gaps

A significant limitation in Flavell's model is the absence of explicit working memory mechanisms for storing information across monitoring cycles. The model does not specify where $\mathcal{ME}^\tau_{\text{difficulty}}$ resides during strategy execution, how $\mathcal{CO}_\tau$ is maintained during evaluation, or how experience patterns across cycles are retained for subsequent processing. This absence precludes sophisticated termination criteria that would require access to historical monitoring data across the complete sequence $\Phi = (\Phi_0, \Phi_1, \ldots, \Phi_T)$.

With an explicit memory component, the model could implement comprehensive abandonment criteria that evaluate: (1) repeated strategy failures indicated by consistently negative $\mathcal{MO}_\tau$ across multiple cycles, suggesting task intractability; (2) resource constraints where cumulative effort across Φ_0 to Φ_τ exceeds acceptable limits relative to $\mathcal{MK}_{\text{Agent}}$; (3) goal displacement where evolving $\mathcal{ME}^{\tau}_{\text{evaluative}}$ signals that alternative objectives have become more salient than the original $\mathcal{G}$; and (4) insurmountable goal-state discrepancy where the pattern of $\mathcal{CO}_\tau$ outcomes reveals fundamental incompatibility with $\mathcal{G}$ achievement.

A related temporal limitation concerns metacognitive knowledge acquisition and refinement. While Flavell acknowledges that experiences can 'add to', 'delete from', or 'revise' the knowledge base, the model assumes pre-existing $\mathcal{MK}$ without specifying learning mechanisms – how unsuccessful strategies refine strategy knowledge, or how repeated encounters improve task assessments.

Such memory-dependent termination decisions and learning-dependent knowledge refinement would better reflect real-world metacognitive monitoring, where individuals track cumulative progress patterns and recognise when persistence becomes counterproductive, while simultaneously refining their metacognitive knowledge through experience. These limitations point towards the necessity for more sophisticated architectural frameworks that explicitly model the temporal dynamics of metacognitive information storage and retrieval as well as the acquisition and refinement of metacognitive knowledge – considerations that become central to Nelson and Narens' metamemory architecture.

3.2 Nelson and Narens' Model of Metamemory

Nelson and Narens [1990] establish metacognition as fundamentally hierarchical, distinguishing between object-level processes that operate on mental content and meta-level processes that operate on cognitive processes themselves. The meta-level maintains a dynamic internal representation of the object-level, enabling self-regulation through two distinct information flows: control (meta-level $\rightarrow$ object-level) and monitoring (object-level $\rightarrow$ meta-level). These relationships are logically independent and asymmetric – the meta-level maintains a model of the object-level while the object-level operates without corresponding meta-level representation. We present the core computational structure below, with complete mathematical formalisation provided in Appendix B.

3.2.1 Acquisition Process

The acquisition process begins with establishing the *norm of study* $\mathcal{N}_s = \rho^* \times (1 + \delta_{\text{retention}})$, where ρ^* represents target performance and $\delta_{\text{retention}}$ captures beliefs about memory decay over interval τ_{delay}. This operationalises abstract goals into quantified mastery criteria that anticipate forgetting. Following Ericsson and Simon [1984], monitoring occurs within working memory (STM), with information from long-term memory (LTM) accessed probabilistically via $\text{retrieve}_\theta(\cdot)$ where θ represents access probability [Atkinson and Shiffrin, 1968].

The **monitoring** phase generates metacognitive experiences as multidimensional vectors. Ease of Learning (EOL) provides initial difficulty assessment, while Feeling of Knowing (FOK) incorporates prior outcomes to refine mastery judgements. These phenomenological experiences serve as primary input for control decisions [Nelson and Narens, 1990, p. 160]. During the **generation** phase, resource allocation operates inversely to EOL/FOK values – items with lower metacognitive confidence receive proportionally more resources $r_{\tau,j} = R_{\text{total}} \times w_j / \sum_k w_k$ where $w_j = (\mathcal{ME}_{\tau,j}[1])^{-1}$. Strategy selection integrates these metacognitive inputs to map appropriate learning methods to individual items.

The **verification** phase employs Judgements of Learning (JOL) to evaluate mastery following cognitive outcomes. Items achieving the norm of study ($\text{JOL}_{\tau,j} \geq \mathcal{N}_s$) are removed from further consideration, whilst those below threshold remain in $\mathcal{J}_{\tau+1}$ for continued learning. The complete experience tuple accumulates in working memory as Φ^{STM}_τ, subsequently undergoing consolidation to LTM at encoding rate ψ.

Algorithm 2 Nelson and Narens: Acquisition

Initialise: $\mathcal{MK}_0^{\text{STM}} \leftarrow \text{retrieve}_\theta(\mathcal{MK}, \mathcal{T}, \mathcal{G})$
$\mathcal{N}_s \leftarrow \rho^* \times (1+ \text{formulate}(\mathcal{MK}_0^{\text{STM}}, \tau_{\text{delay}}, \mathcal{T}, \mathcal{G}))$
$\mathcal{J}_0 \leftarrow \{1, \ldots, N\}$; $\tau \leftarrow 1$; $\Phi_0^{\text{STM}} \leftarrow \emptyset$
while $\mathcal{J}_\tau \neq \emptyset$ **do**
  // MONITOR: Assess mastery via EOL/FOK
  $\mathcal{MK}_\tau^{\text{STM}} \leftarrow \mathcal{MK}_{\tau-1}^{\text{STM}} \cup \text{retrieve}_\theta(\mathcal{MK}, \mathcal{ME}_{\tau-1})$
  for each $j \in \mathcal{J}_\tau$ **do**
    $\mathcal{ME}_{\tau,j}[1] \leftarrow$ **if** $\tau = 1$ **then** $\text{EOL}(i_j)$ **else** $\text{FOK}(i_j, \mathcal{CO}_{\tau-1,j})$
  end for
  // GENERATE: Allocate resources & select strategies
  for each $j \in \mathcal{J}_\tau$ **do**
    $r_{\tau,j} \leftarrow R_{\text{total}} \times (\mathcal{ME}_{\tau,j}[1])^{-1} / \sum_k (\mathcal{ME}_{\tau,k}[1])^{-1}$
    $\sigma_{\tau,j} \leftarrow \text{select}(s \in \mathcal{MK}_{\text{Strategy}} \mid i_j, r_{\tau,j}, \mathcal{ME}_{\tau,j})$
    $\mathcal{CO}_{\tau,j} \leftarrow \text{execute}(i_j, r_{\tau,j}, \sigma_{\tau,j})$
  end for
  // VERIFY: Judge learning & update items
  for each $j \in \mathcal{J}_\tau$ **do**
    $\text{JOL}_{\tau,j} \leftarrow \text{feel}(i_j, \mathcal{CO}_{\tau,j}) \oplus \text{assess}(i_j, \mathcal{CO}_{\tau,j}, \mathcal{MK}_\tau^{\text{STM}})$
    $\mathcal{ME}_{\tau,j}[2] \leftarrow \text{JOL}_{\tau,j}$
    $\Phi_\tau^{\text{STM}} \leftarrow \Phi_\tau^{\text{STM}} \cup \{(\mathcal{ME}_{\tau,j}, i_j, r_{\tau,j}, \sigma_{\tau,j}, \mathcal{CO}_{\tau,j})\}$
  end for
  $\mathcal{J}_{\tau+1} \leftarrow \{j \in \mathcal{J}_\tau : \mathcal{N}_s - \text{JOL}_{\tau,j} > 0\}$; $\tau \leftarrow \tau + 1$
end while
$\mathcal{MK} \leftarrow \text{consolidate}_\psi(\mathcal{MK}, \Phi_\tau^{\text{STM}})$ ▷ Experience to LTM

3.2.2 Retrieval Process

The retrieval process implements Nelson and Narens' dual-counter FOK hypothesis, where FOK^+ accumulates evidence for information presence whilst FOK^- accumulates evidence for absence, consistent with 'knowing not' [Kolers and Palef, 1976]. Initial thresholds are personalised through metacognitive calibration history: $\lambda_{\text{FOK}}^{(0)} = \text{median}(\{||\text{FOK}|| : \text{successful retrievals in } \mathcal{MK}_0^{\text{STM}}\})$ and $\lambda_{\text{confidence}}^{(0)} = \text{median}(\{\text{confidence} : \text{correct outputs in } \mathcal{MK}_0^{\text{STM}}\})$, embodying the No-Magic Hypothesis by utilising recallable metacognitive knowledge.

The **monitoring** phase employs rapid FOK assessment that operates faster than actual recall [Reder, 1987], enabling efficient search control. When FOK magnitude falls below threshold ($||\text{FOK}_\tau|| < \lambda_{\text{FOK}}^{(\tau)}$), insufficient evidence triggers intensive cue attention to gather additional metacognitive information. With sufficient evidence, positive dominance ($\text{FOK}_\tau^+ > \text{FOK}_\tau^-$) warrants continued search, while negative dominance justifies termination.

The **generation** phase reflects Nelson and Narens' insight that search execution is automatic once initiated – *conscious control operates through cue attention intensity rather than strategy selection*. The automatic search process $\text{search}_{\text{auto}}(\text{cue}_\tau)$ operates through pattern recognition, potentially yielding identical results across consecutive cycles due to its deterministic nature.

Verification distinguishes two error pathways: commission errors (outputting incorrect answers with high confidence) and omission errors (terminating without answers following prolonged search). Following satisficing principles [Simon, 1979], both confidence and FOK thresholds undergo dynamic adjustment: $\lambda^{(\tau+1)} = \lambda^{(0)} \cdot \beta_\tau$ where $\beta_\tau = \exp(-\alpha \cdot \text{burden})$ captures accumulating search costs. This ensures previously inadequate answers may become acceptable as search burden increases, preventing exhaustive search behaviour.

3.2.3 Memory Consolidation and Knowledge Evolution

A distinctive strength of Nelson and Narens' framework lies in its explicit treatment of long-term memory (LTM) as both a repository and an evolving knowledge base. During acquisition and retrieval,

Algorithm 3 Nelson and Narens: Retrieval

Initialise: $\mathcal{MK}_0^{\text{STM}} \leftarrow \text{retrieve}_\theta(\mathcal{MK}, \mathcal{Q})$
$\lambda_{\text{FOK}}^{(0)}, \lambda_{\text{confidence}}^{(0)} \leftarrow \text{calibrate}(\mathcal{MK}_0^{\text{STM}})$; $\tau \leftarrow 0$; $\Omega_0^{\text{STM}} \leftarrow \emptyset$
while search active **do**
  // MONITOR: Assess dual-counter FOK
  $\mathcal{MK}_\tau^{\text{STM}} \leftarrow \mathcal{MK}_{\tau-1}^{\text{STM}} \cup \text{retrieve}_\theta(\mathcal{MK}, \text{FOK}_{\tau-1})$ if $\tau > 0$
  $[\text{FOK}_\tau^+, \text{FOK}_\tau^-] \leftarrow \text{feel}(\mathcal{Q}, \mathcal{A}_{\tau-1}) \oplus \text{assess}(\mathcal{Q}, \mathcal{A}_{\tau-1}, \mathcal{MK}_\tau^{\text{STM}})$
  // Determine search intensity based on FOK evidence
  if $||\text{FOK}_\tau|| < \lambda_{\text{FOK}}^{(\tau)}$ **then**
    $\mathcal{S}_\tau \leftarrow \text{ACTIVE}_{\text{intensive}}$ ▷ Insufficient evidence
  else if $\text{FOK}_\tau^+ > \text{FOK}_\tau^-$ **then**
    $\mathcal{S}_\tau \leftarrow \text{ACTIVE}_{\text{standard}}$ ▷ Positive dominance
  else
    break ▷ Negative dominance: terminate
  end if
  // GENERATE: Attend to cues & automatic search
  $\text{cue}_\tau \leftarrow \text{attend}_{[\text{intensive/standard}]}(\mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}})$ based on $\mathcal{S}_\tau$
  $\mathcal{A}_\tau \leftarrow \text{search}_{\text{auto}}(\text{cue}_\tau)$ ▷ Automatic pattern recognition
  // VERIFY: Evaluate answer & adjust thresholds
  $\text{confidence}_\tau \leftarrow \text{assess}(\mathcal{A}_\tau, \mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}})$ if $\mathcal{A}_\tau \neq \text{null}$
  if $\mathcal{A}_\tau \neq \text{null} \wedge \text{confidence}_\tau \geq \lambda_{\text{confidence}}^{(\tau)}$ **then**
    output $\mathcal{A}_\tau$; **break**
  else if $\mathcal{A}_\tau = \text{null} \wedge \text{FOK}_\tau^- > \text{FOK}_\tau^+$ **then**
    output null; **break** ▷ Omission
  end if
  $\Omega_\tau^{\text{STM}} \leftarrow \Omega_\tau^{\text{STM}} \cup \{(\text{FOK}_\tau, \text{cue}_\tau, \mathcal{A}_\tau, \text{confidence}_\tau)\}$
  $\beta_\tau \leftarrow \exp(-\alpha \cdot (\tau + |\{\text{failed attempts in } \Omega_\tau^{\text{STM}}\}|))$
  $\lambda_{\text{confidence}}^{(\tau+1)}, \lambda_{\text{FOK}}^{(\tau+1)} \leftarrow \lambda^{(0)} \cdot \beta_\tau$ ▷ Satisficing
  $\tau \leftarrow \tau + 1$
end while
$\mathcal{MK} \leftarrow \text{consolidate}_\psi(\mathcal{MK}, \Omega_\tau^{\text{STM}})$ ▷ Experience to LTM

the experience tuples accumulated in working memory (Ω_T^{STM}) undergoes consolidation into LTM at encoding rate ψ:

While Nelson and Narens do not explicitly specify the timing of this consolidation process, it likely occurs during the verification stage at rate ψ, potentially operating below conscious awareness. This consolidation mechanism enables the global metacognitive knowledge base to evolve through accumulated experience, distinguishing Nelson and Narens' approach from more static metacognitive frameworks. The probabilistic retrieval function $\text{retrieve}_\theta(\mathcal{MK}, \cdot)$ subsequently accesses this enriched knowledge base, creating a dynamic feedback loop where metacognitive experiences inform future metacognitive assessments.

4 Discussion

Our algorithmic formalisations of Flavell's and Nelson and Narens' frameworks offer one possible interpretation of reasoning system limitations. The prefix dominance trap might be understood as reflecting absent pre-generation monitoring – without FOK assessments and strategy selection, systems cannot evaluate approaches before committing. Hallucination could correspond to Nelson and Narens' commission errors, suggesting missing confidence thresholds or termination criteria. Inconsistent self-evaluation may indicate the absence of hierarchical monitoring structures that maintain coherence in human metacognition. While these formalisations provide structured ways to think about computational components – monitoring loops, confidence transformations, working memory structures – we acknowledge that psychological theories developed for human cognition may not map cleanly onto artificial systems. The following specifications suggest architectural considerations rather than definitive solutions, and considerable work remains to determine whether these cognitive principles can meaningfully guide computational implementation.

Dynamic Memory-Guided Reasoning Against Hallucination Nelson and Narens' $\lambda_{\text{confidence}}$ mechanism suggests a theoretical approach to hallucination through coordinated memory-reasoning confidence assessment. Their framework implies that systems might maintain domain-specific thresholds: $\lambda_{\text{confidence}}^{\text{(domain)}} = \text{median}(\{\text{confidence}_\tau : \text{correct outputs in domain}\})$, calibrated from performance history. When confidence falls below $\lambda_{\text{confidence}}^{(\tau)}$, Nelson and Narens' model prescribes continuing search rather than outputting uncertain answers – directly addressing the commission error problem underlying hallucination. The dynamic adjustment mechanism $\beta_\tau = \exp(-\alpha \cdot \text{search burden})$ suggests how systems might balance accuracy against computational cost, maintaining high thresholds for factual domains while adapting to task demands. This distinction between commission errors (hallucination) and omission errors (refusing to answer) provides theoretical grounding for confidence-based output control.

Attention Coordination Through Metacognitive Monitoring The finding that LLMs operate within a lower-dimensional "metacognitive space" [Ji-An et al., 2025] resonates with multiple theoretical predictions. Flavell's $\mathcal{ME}_{\text{difficulty}}^{\tau}$ formulation suggests hierarchical monitoring of cognitive states, while Nelson and Narens' FOK mechanism implies rapid accessibility assessment operating faster than full memory retrieval – consistent with Reder's [1987, 1988] findings that FOK judgments have shorter latency than actual recall. This convergence suggests that efficient metacognitive monitoring might naturally operate on compressed representations of cognitive state. The theoretical framework indicates how different reasoning processes might require different levels of monitoring granularity, potentially explaining why models report some computations accurately while failing on others [Lindsey et al., 2025]. These insights, combined with findings that implicit confidence measures show greater metacognitive sensitivity than explicit ones [Xiong et al., 2023], suggest that attention allocation could benefit from monitoring internal representations rather than relying solely on verbalized assessments.

Strategic Reasoning Selection via Metacognitive Control MeCo's dual-threshold policy [Li et al., 2025] parallels both Flavell's strategy selection mechanism and Nelson and Narens' dual-counter FOK hypothesis. Flavell's formulation $\mathcal{CS}_\tau = \text{select}(s \in \mathcal{MK}_{\text{Strategy}} \mid \mathcal{ME}_{\text{difficulty}}^{\tau}, \mathcal{MK}_\tau, \mathcal{T}, \mathcal{G})$ suggests how metacognitive experiences guide strategy choice. Nelson and Narens' use of FOK magnitude $||\text{FOK}_\tau||$ for search intensity decisions – intensive attention when $||\text{FOK}_\tau|| < \lambda_{\text{FOK}}^{(\tau)}$ versus standard or termination based on counter dominance – provides a principled mechanism for adaptive control. MeCo's implementation of weak versus strong metacognitive signals, detected through linearly separable directions in representation space [Zou et al., 2023, Liu et al., 2023], mirrors this magnitude-based approach. This convergence indicates how reasoning strategy selection – whether chain-of-thought, direct retrieval, or tool invocation – could be guided by formal metacognitive principles rather than heuristic rules.

Memory Consolidation for Adaptive Learning Nelson and Narens' experience consolidation mechanism ($\Phi_\tau^{\text{STM}} \xrightarrow{\psi} \text{LTM}$) pinpoints a fundamental gap in current LLM capabilities. Their formulation of experience tuples $\Phi_\tau = (\mathcal{ME}_\tau, \text{Strategy}_\tau, \text{Outcome}_\tau)$ suggests how systems might learn from their own cognitive experiences, updating metacognitive knowledge about strategy effectiveness, monitoring accuracy, and resource requirements. This aligns with Didolkar et al. [2024]'s extraction of metacognitive knowledge through skill annotations, though current approaches capture only static snapshots rather than dynamic learning. The framework's emphasis on pattern recognition across experience tuples – identifying which strategies succeed for which task types under which confidence levels – suggests that true adaptive behavior requires explicit mechanisms for metacognitive memory beyond current in-context learning. While current LLMs rely on fixed architectures with static learning mechanisms, concepts like the Darwin Gödel Machine [Zhang et al., 2025] point toward systems that could empirically evolve their own metacognitive strategies, transforming the consolidation process itself based on accumulated experience.

5 Limitations

While our formalisation provides theoretical foundations for understanding reasoning failures through metacognitive principles, several limitations constrain the immediate applicability of this work.

Implementation challenges We have not proposed how to implement metacognitive constructs such as $\mathcal{ME}_{\text{difficulty}}$ or $\text{FOK}^{+/-}$ in practice. Whether these subjective "feelings" should be realised as learned representations, designed features, or emerge from architectural properties remains unspecified. The translation from psychological experiences to computational mechanisms presents non-trivial challenges, particularly given the parallel nature of transformer architectures versus the sequential processing assumed in our algorithms.

Computational complexity The MGV framework introduces substantial overhead through iterative monitoring-generation-verification cycles, multiple metacognitive state updates, and experience consolidation mechanisms. We have not analysed the computational cost-benefit trade-offs, nor determined whether simpler metacognitive mechanisms might achieve comparable benefits with reduced complexity.

Theoretical foundations We largely accept Flavell's and Nelson and Narens' theories without critical examination of their limitations or controversies within cognitive science. Alternative metacognitive theories might provide different insights, and the suitability of these particular frameworks for computational systems remains unexamined.

6 Conclusion

This paper has presented a theoretical exploration of how metacognitive principles from cognitive science might address fundamental limitations in language model reasoning architectures. Through systematic formalisation of Flavell's cognitive monitoring model and Nelson and Narens' metamemory framework, we have translated psychological constructs into computational structures, revealing specific architectural gaps in current test-time reasoning approaches. The resulting Monitor-Generate-Verify framework suggests how explicit metacognitive mechanisms – pre-generation monitoring, dynamic confidence thresholds, and experience consolidation – could potentially address issues such as the prefix dominance trap.

However, translating these theoretical insights into practical architectures remains an open challenge. The gap between abstract metacognitive experiences and implementable computational mechanisms is substantial, and whether the additional complexity of MGV architectures would yield commensurate benefits requires empirical investigation. Future work must address how to operationalise subjective monitoring signals, implement working memory mechanisms compatible with parallel processing, and design efficient consolidation processes that enable genuine metacognitive learning.

Despite these limitations, we believe examining established cognitive theories provides valuable perspectives on architectural evolution. As language models increasingly tackle complex reasoning tasks, understanding the computational principles underlying human metacognition may guide development of systems that monitor their own cognitive states, adaptively select strategies, and learn from their reasoning experiences. Whether through direct implementation of MGV principles or through architectures inspired by metacognitive insights, bridging cognitive science and artificial intelligence remains a promising direction for advancing reasoning capabilities.

References

Richard C Atkinson and Richard M Shiffrin. Human memory: A proposed system and its control processes. In *Psychology of learning and motivation*, volume 2, pages 89–195. Elsevier, 1968.

Harry P Bahrick. Two-phase model for prompted recall. *Psychological Review*, 77(3):215, 1970.

Trent N Cash, Daniel M Oppenheimer, Sara Christie, and Mira Devgan. Quantifying uncertainty: Testing the accuracy of llms' confidence judgments. *PsyArXiv 47df5_v3*, 10, 2024.

Aniket Didolkar, Anirudh Goyal, Nan Rosemary Ke, Siyuan Guo, Michal Valko, Timothy Lillicrap, Danilo Jimenez Rezende, Yoshua Bengio, Michael C Mozer, and Sanjeev Arora. Metacognitive capabilities of llms: An exploration in mathematical problem solving. *Advances in Neural Information Processing Systems*, 37:19783–19812, 2024.

K. Anders Ericsson and Herbert A. Simon. *Protocol analysis: Verbal reports as data*. The MIT Press, 1984.

John H. Flavell. *The developmental psychology of Jean Piaget*. D Van Nostrand, 1963. doi: 10.1037/11449-000.

John H Flavell. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American Psychologist*, 34(10):906, 1979.

Stephen M Fleming. Metacognition and confidence: A review and synthesis. *Annual Review of Psychology*, 75(1):241–268, 2024.

Maxime Griot, Coralie Hemptinne, Jean Vanderdonckt, and Demet Yuksel. Large language models lack essential metacognition for reliable medical reasoning. *Nature communications*, 16(1):642, 2025.

Li Ji-An, Hua-Dong Xiong, Robert C Wilson, Marcelo G Mattar, and Marcus K Benna. Language models are capable of metacognitive monitoring and control of their internal activations. *arXiv preprint arXiv:2505.13763*, 2025.

Walter Kintsch. More on recognition failure of recallable words: Implications for generation–recognition models. *Psychological Review*, 85(5):470, 1978.

P. A. Kolers and S. R. Palef. Knowing not. *Memory & Cognition*, 4(5):553–558, 1976. doi: 10.3758/BF03213218.

Hyunseok Lee, Seunghyuk Oh, Jaehyung Kim, Jinwoo Shin, and Jihoon Tack. Revise: Learning to refine at test-time via intrinsic self-verification. *arXiv preprint arXiv:2502.14565*, 2025.

Wenjun Li, Dexun Li, Kuicai Dong, Cong Zhang, Hao Zhang, Weiwen Liu, Yasheng Wang, Ruiming Tang, and Yong Liu. Adaptive tool use in large language models with meta-cognition trigger. *arXiv preprint arXiv:2502.12961*, 2025.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, et al. On the biology of a large language model. *Transformer Circuits Thread*, 2025.

Wenhao Liu, Xiaohua Wang, Muling Wu, Tianlong Li, Changze Lv, Zixuan Ling, Jianhao Zhu, Cenyuan Zhang, Xiaoqing Zheng, and Xuanjing Huang. Aligning large language models with human preferences through representation engineering. *arXiv preprint arXiv:2312.15997*, 2023.

Tongxu Luo, Wenyu Du, Jiaxi Bi, Stephen Chung, Zhengyang Tang, Hao Yang, Min Zhang, and Benyou Wang. Learning from peers in reasoning models. *arXiv preprint arXiv:2505.07787*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023.

Thomas O Nelson and Louis Narens. Metamemory: A theoretical framework and new findings. In *Psychology of learning and motivation*, volume 26, pages 125–173. Elsevier, 1990.

Lynne M. Reder. Strategy selection in question answering. *Cognitive Psychology*, 19(1):90–138, 1987.

Lynne M. Reder. Strategic control of retrieval strategies. In *Psychology of Learning and Motivation*, volume 22, pages 227–259. Academic Press, 1988.

Herbert A. Simon. *Models of Thought*, volume 1. Yale University Press, 1979.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*, 2023.

Guoqing Wang, Wen Wu, Guangze Ye, Zhenxiao Cheng, Xi Chen, and Hong Zheng. Decoupling metacognition from cognition: A framework for quantifying metacognitive ability in llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25353–25361, 2025.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large language models are better reasoners with self-verification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2550–2575, 2023.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin godel machine: Open-ended evolution of self-improving agents. *arXiv preprint arXiv:2505.22954*, 2025.

Yunxiang Zhang, Muhammad Khalifa, Lajanugen Logeswaran, Jaekyeom Kim, Moontae Lee, Honglak Lee, and Lu Wang. Small language models need strong verifiers to self-correct reasoning. *arXiv preprint arXiv:2404.17140*, 2024.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Flavell's Model of Cognitive Monitoring

Flavell [1979] describes metacognition as a dynamic control architecture comprising four interacting components: *metacognitive knowledge*, *metacognitive experience*, *goals* (or tasks), and *actions* (or strategies). Rather than viewing metacognition as merely stored knowledge about cognition, Flavell presents it as a dynamic control system. This system operates through continuous interactions between four elements: what agents know about their cognitive capabilities (metacognitive knowledge), what they feel about their current cognitive state (metacognitive experiences), what they aim to achieve (goals), and how they control their thinking (strategies). Central to Flavell's model is the principle of reciprocal interaction amongst these components. Rather than operating as independent modules, they form an integrated system characterised by continuous bidirectional influences: metacognitive knowledge guides both strategy selection and the interpretation of ongoing cognitive experiences; these conscious experiences, in turn, update the knowledge base and prompt strategic adjustments; task-goals determine which aspects of metacognitive knowledge become most salient; and the outcomes of chosen actions provide feedback that shapes both immediate metacognitive experiences and longer-term understanding of effective cognitive approaches. This dynamic interplay positions metacognition as a self-regulating system capable of adaptive control over cognitive processes.

Initialisation Let $\mathcal{T}$ be a task and $\mathcal{G}$ be the associated goal. We establish the initial system state:

$$\mathcal{S}_0 = f(\mathcal{T}, \mathcal{G})$$

where $(\mathcal{T}, \mathcal{G})$ is self-imposed or externally-imposed.

While Flavell [1979] treats 'goals' and 'tasks' as equivalent, we maintain a computational distinction to enhance the model's precision: $\mathcal{T}$ represents the specific cognitive enterprise, whilst $\mathcal{G}$ represents the desired outcome or success criteria. This separation enables clearer analysis of metacognitive processes, such as assessing the cognitive demands of $\mathcal{T}$ relative to $\mathcal{G}$, or specifying which approaches to employ for $\mathcal{T}$ to achieve $\mathcal{G}$. For instance, the same reasoning task ($\mathcal{T}$: logical problem-solving) might require different metacognitive assessments depending on whether the goal is speed ($\mathcal{G}_1$: quick approximation) or accuracy ($\mathcal{G}_2$: verified solution).

M-G-V (Information Processing) Cycle For monitoring cycles $\tau = 0, 1, \ldots, T$:

WHILE $\mathcal{S}_\tau =$ ACTIVE:

1. **MONITOR**: *Monitor cognitive status through retrieval of metacognitive knowledge and assessment of metacognitive experience*.

   Knowledge activation operates differently across metacognitive cycles, with initial cycles relying solely on task-goal combinations while ongoing cycles incorporate emerging metacognitive experiences. At $t = 0$, the system identifies potentially relevant metacognitive knowledge based exclusively on the task-goal pairing. In subsequent cycles ($\tau > 0$), the knowledge base expands as metacognitive experiences ($\mathcal{ME}$) from the previous cycle $\tau - 1$ triggering additional relevant knowledge.

$$\mathcal{MK}_\tau = \begin{cases} \text{retrieve}(\mathcal{MK}, \mathcal{T}, \mathcal{G}) & \text{if } \tau = 0 \\ \mathcal{MK}_{\tau-1} \cup \text{retrieve}(\mathcal{MK}, \mathcal{ME}_{\tau-1}) & \text{if } \tau > 0 \end{cases}$$

   According to Flavell [1979], metacognitive knowledge comprises three major categories:

   - **Agent Variables** ($\mathcal{MK}_{\text{Agent}}$): Knowledge about cognitive agents' characteristics and capabilities that applies across different cognitive endeavours. These are fundamentally subjective beliefs about processing preferences, strengths, and limitations rather than objective assessments. For computational agents, these may represent *learned self-models* – representations of performance patterns, processing preferences, and comparative capabilities derived from experience across cognitive tasks.
   - **Task Variables** ($\mathcal{MK}_{\text{Task}}$): Knowledge about cognitive situation assessment, including: (1) information characteristics (e.g., familiarity, complexity, organisation), and (2) task demands and goals. This knowledge is evaluative – understanding what task characteristics mean for cognitive processes and goal achievement, not merely recognising the characteristics themselves.

- **Strategy Variables** ($\mathcal{MK}_{\text{Strategy}}$): Knowledge concerning the effectiveness of cognitive strategies ($\mathcal{CS}$) and metacognitive strategies ($\mathcal{MS}$). Across different goals and task types, $\mathcal{CS}$ are cognitive operations that address problem-solving procedures such as applying domain-specific algorithms or step-by-step problem decomposition, whereas $\mathcal{MS}$ monitor and regulate such cognitive processes. For instance, chain-of-thought reasoning represents a $\mathcal{CS}$ for solving problems systematically, while deciding to *employ* chain-of-thought based on problem complexity assessment represents a $\mathcal{MS}$. Flavell [1979] explicitly incorporates both strategy types within this category, reflecting his theoretical position that strategy selection constitutes a fundamentally metacognitive process requiring knowledge about when, how, and why particular approaches prove effective under specific conditions.

These categories function as an integrated system: task variables diagnose cognitive demands, strategy variables prescribe responses, and agent variables contextualise both within the agent's capabilities[1].

Flavell [1979] distinguishes between knowledge-based experiences, which 'are best described as items of metacognitive knowledge that have entered consciousness' (e.g., suddenly recalling a relevant strategy), and feeling-based experiences, which 'clearly cannot be described that way' (e.g., feeling confused).

This dual nature of metacognitive experience – alternating between immediate phenomenological feelings and knowledge-based assessments – motivates our formal representation using the exclusive-or operator $\oplus$. In this formulation, $feel()$ captures the pure subjective sensations of cognitive state, while $assess()$ represents evaluations informed by metacognitive knowledge. The operator $\oplus$ thus reflects Flavell's distinction between feeling-based experiences (phenomenological states that cannot be reduced to knowledge) and knowledge-based experiences (instances of metacognitive knowledge entering consciousness).

Our exclusive-or formalisation captures the observation that these two modes typically alternate rather than blend, though we acknowledge that this binary representation constitutes a modelling simplification of potentially richer interactions. Accordingly, this binary characterisation suggests that, at any given moment, an agent experiences either raw cognitive feelings awaiting interpretation or automatic, knowledge-influenced assessments. The temporal alternation between these exclusive states gives rise to the evolving metacognitive experience that guides subsequent processing.

At this stage, it is notable that Flavell [1979, p. 909] primarily associates metacognitive experience with the subjective sense of perceived difficulty. Such experiences may involve feelings of complexity, comprehension challenges, conceptual opacity, or the sense that material exceeds current capabilities.

$$\mathcal{ME}^{\tau}_{\text{difficulty}} = \begin{cases} \text{feel}(\mathcal{T}) \oplus \text{assess}(\mathcal{T}, \mathcal{MK}_{\tau}) & \text{if } \tau = 0 \\ \text{feel}(\mathcal{T}, \text{Outcomes}_{\tau-1}) \oplus \text{assess}(\mathcal{T}, \text{Outcomes}_{\tau-1}, \mathcal{MK}_{\tau}) & \text{if } \tau > 0 \end{cases}$$

Accordingly, $\mathcal{ME}^{\tau}_{\text{difficulty}}$ evolves through iterative cycle-dependent assessments, progressing from initial coarse-grained feelings to increasingly nuanced evaluations that identify specific challenge sources and their implications for strategy selection. These experiences could help identify specific sources of obstacles and serve to guide the agent's attentional and regulatory focus.

2. **GENERATE**: *Control cognitive activity through strategy selection and execution.*

   Flavell [1979, p. 909] emphasises that this stage centres on the selection of cognitive strategies ($\mathcal{CS}_{\tau}$) through the integration of metacognitive experiences and knowledge. Metacognitive experiences of difficulty ($\mathcal{ME}_{\text{difficulty}}$), whether feeling-based or knowledge-based, function as computational *signals* that indicate cognitive status. However, these signals require interpretation through metacognitive knowledge to guide effective strategy selection.

[1] The distinction between $\mathcal{MK}_{\text{Strategy}}$ and $\mathcal{MK}_{\text{Task}}$ emerges from their functional roles. $\mathcal{MK}_{\text{Task}}$ enables diagnosis by identifying what makes cognitive enterprises demanding and how task characteristics influence goal achievement probability, whereas $\mathcal{MK}_{\text{Strategy}}$ enables prescription by specifying which cognitive approaches to deploy given those diagnostic assessments. Task variables answer 'what challenges does $\mathcal{T}$ present relative to $\mathcal{G}$?' and strategy variables answer 'which approaches for $\mathcal{T}$ will achieve $\mathcal{G}$?'

As Flavell [1979, p. 906] establishes, effective cognitive regulation emerges only when metacognitive experiences combine with metacognitive knowledge, transforming ambiguous feelings into actionable strategic decisions.

The strategy selection process draws upon $\mathcal{MK}_{\text{Strategy}}$, which encompasses knowledge about both metacognitive strategies ($\mathcal{MS}$) and cognitive strategies ($\mathcal{CS}$). Although Flavell does not specify the exact selection mechanism, his examples suggest a *two-phase pattern-matching* process. In the first phase, $\mathcal{MK}_{\text{Strategy}}$ guides the interpretation of $\mathcal{ME}^{\tau}_{\text{difficulty}}$, transforming general difficulty signals into precise diagnostic patterns. For instance, metacognitive knowledge might specify "when experiencing content uncertainty, identify specific unknown terms" or "when procedurally confused, assess whether confusion stems from missing steps versus unclear sequence". In the second phase, these refined difficulty patterns activate corresponding cognitive strategies from $\mathcal{MK}_{\text{Strategy}}$ – content uncertainty with identified terms triggers seeking definitions, whilst procedural confusion from missing steps activates searching for worked examples. Throughout this process, the selection mechanism integrates agent capabilities ($\mathcal{MK}_{\text{Agent}}$) and task characteristics ($\mathcal{MK}_{\text{Task}}$) to identify the most appropriate strategy for achieving $\mathcal{G}$ given $\mathcal{T}$.

$$\mathcal{CS}_{\tau} = \text{select}(s \in \mathcal{MK}_{\text{Strategy}} \mid \mathcal{ME}^{\tau}_{\text{difficulty}}, \mathcal{MK}_{\tau}, \mathcal{T}, \mathcal{G})$$

The selected cognitive strategy is implemented to produce cognitive outcomes ($\mathcal{CO}_{\tau}$), generating a feedback that is rich in nature, as as it encompasses not only task progress information but also a new context for the next cycle's monitoring and potential strategy adjustment.

$$\mathcal{CO}_{\tau} = \text{execute}(\mathcal{CS}_{\tau}, \mathcal{T}, \mathcal{G})$$

3. **VERIFY**: *Evaluate progress and determine continuation.*

Following strategy execution, Flavell [1979, p. 909] comments that the outcomes potentially 'trigger additional metacognitive experiences about how the endeavour is faring'. These evaluative experiences ($\mathcal{ME}^{\tau}_{\text{evaluative}}$) are about performance rather than difficulty.

$$\mathcal{ME}^{\tau}_{\text{evaluative}} = \text{feel}(\mathcal{CO}_{\tau}) \oplus \text{assess}(\mathcal{CO}_{\tau}, \mathcal{MK}_{\tau})$$

These experiences, again informed and guided by pertinent metacognitive knowledge, instigate the metacognitive strategy of surveying 'all that [the agent has] learned to see if it fits together into a coherent whole, if it seems plausible and consistent with [the agent's] prior knowledge and expectations, and if it provides an avenue to the goal' [Flavell, 1979, p. 909].

$$\mathcal{MS}_{\tau} = \text{select}(s \in \mathcal{MK}^{\text{meta}}_{\text{Strategy}} \mid \mathcal{ME}^{\tau}_{\text{evaluative}}, \mathcal{MK}_{\tau}, \mathcal{CO}_{\tau}, \mathcal{G})$$

where

$$\mathcal{MS}_{\tau} = \begin{cases} \text{coherence} & \text{if } \mathcal{ME}^{\tau}_{\text{evaluative}} \text{ signals fragmented understanding} \\ \text{plausibility} & \text{if } \mathcal{ME}^{\tau}_{\text{evaluative}} \text{ signals doubtful results} \\ \text{consistency} & \text{if } \mathcal{ME}^{\tau}_{\text{evaluative}} \text{ signals unexpected outcomes} \\ \text{goal-conduciveness} & \text{if } \mathcal{ME}^{\tau}_{\text{evaluative}} \text{ signals uncertain progress} \end{cases}$$

$\mathcal{ME}^{\tau}_{\text{evaluative}}$ signals the need for assessment. For example, "feeling uncertain about validity of the processed outcome" or "sensing incomplete understanding despite completion of process". These evaluative experiences activate relevant metacognitive strategies from $\mathcal{MK}_{\text{Strategy}}$. For instance, uncertainty about validity triggers plausibility checking, while sensing incompleteness activates coherence assessment to identify gaps.

$\mathcal{MS}_{\tau}$ represents the strategic choice to conduct comprehensive evaluation along four possible dimensions: coherence ("do the outcomes form a consistent understanding?"), plausibility ("are the results believable given prior knowledge?"), consistency ("do outcomes align with initial expectations?"), and goal-conduciveness ("do current results provide a pathway to goal achievement?"). The execution systematically evaluates $\mathcal{CO}_{\tau}$ against relevant knowledge:

$$\mathcal{MO}_\tau = \text{execute}(\mathcal{MS}_\tau, \mathcal{CO}_\tau, \mathcal{MK}_\tau, \mathcal{G})$$

Flavell emphasises that metacognitive experiences can 'add to', 'delete from', or 'revise' the metacognitive knowledge base through Piagetian [Flavell, 1963] mechanisms. The agent observes relationships among goals, strategies, experiences, and outcomes across the complete monitoring cycle.

Let Φ_τ represents the complete experience tuple, where $\mathcal{ME}_\tau = (\mathcal{ME}^\tau_{\text{difficulty}}, \mathcal{ME}^\tau_{\text{evaluative}})$, $\text{Strategy}_\tau = (\mathcal{CS}_\tau, \mathcal{MS}_\tau)$ and $\text{Outcome}_\tau = (\mathcal{CO}_\tau, \mathcal{MO}_\tau)$:

$$\Phi_\tau = (\mathcal{ME}_\tau, \text{Strategy}_\tau, \text{Outcome}_\tau) \tag{1}$$

$$\mathcal{MK} = \text{update}(\mathcal{MK}, \Phi_\tau) \tag{2}$$

Based on the comprehensive metacognitive evaluation, the system determines its next state:

$$\mathcal{S}_{\tau+1} = \begin{cases} \text{ACTIVE} & \text{if } \neg\text{goal\_achieved}(\mathcal{CO}_\tau, \mathcal{G}) \\ \text{TERMINATE} & \text{if goal\_achieved}(\mathcal{CO}_\tau, \mathcal{G}) \end{cases}$$

B Nelson and Narens' Model of Metamemory

Nelson and Narens [1990] theorise metacognitive systems with particular focus on metamemory in the context of self-directed, self-paced learning and retrieval tasks. Their framework establishes metacognition as fundamentally hierarchical, distinguishing between cognitive processes that operate on mental content (object-level) and those that operate on cognitive processes themselves (meta-level). This two-level architecture provides the theoretical foundation for understanding how cognitive systems achieve self-regulation and control during learning activities.

According to their model, the meta-level maintains a dynamic internal representation of the object-level, functioning as a mental simulation that enables the system to monitor current cognitive states and guide transitions towards desired goals. The interaction between levels operates through two distinct information flows, *control* (meta-level $\rightarrow$ object-level) and *monitoring* (object-level $\rightarrow$ meta-level). Control processes enable the meta-level to modify object-level states or processes – such as allocating study time to difficult material or switching from rote memorisation to elaborative rehearsal strategies. Monitoring processes provide the meta-level with information about current object-level states, updating its internal model of the cognitive situation. These relationships connotes two notable properties: they are logically independent (control does not inherently generate feedback about its effects) and asymmetric (the meta-level maintains a model of the object-level whilst the object-level operates without any corresponding representation of the meta-level).

B.1 Acquisition Process

Initialisation Given a task ($\mathcal{T}$) and goal ($\mathcal{G}$) with a target performance level (ρ^*), the agent establishes the *norm of study* ($\mathcal{N}_s$):

$$\begin{aligned} \mathcal{MK}_0^{\text{STM}} &= \text{retrieve}_\theta(\mathcal{MK}, \mathcal{T}, \mathcal{G}) \\ \delta_{\text{retention}} &= \text{formulate}(\mathcal{MK}_0^{\text{STM}}, \tau_{\text{delay}}, \mathcal{T}, \mathcal{G}) \\ \mathcal{N}_s &= \rho^* \times (1 + \delta_{\text{retention}}) \end{aligned}$$

At the initialisation stage ($\tau = 0$), a global metacognitive parameter ($\mathcal{N}_s$) operationalises abstract goals into quantified mastery criteria, which Nelson and Narens [1990, p. 130] define as 'the overall degree of mastery the person believes should be attained during acquisition'.

Following Ericsson and Simon [1984], monitoring operations occur within working memory (STM), with $\mathcal{MK}_0^{\text{STM}}$ denoting the metacognitive knowledge retrieved into this workspace at $\tau = 0$. Information from long-term memory (LTM) may be accessed by first copying it into STM with probability θ [Atkinson and Shiffrin, 1968], captured through the notation $\text{retrieve}_\theta(\cdot)$ for this probabilistic access during metacognitive monitoring. The term $\delta_{\text{retention}}$ represents the agent's theory of retention – beliefs about memory decay over the interval τ_{delay}.

This formulation reflects Nelson and Narens' insight that effective learning requires anticipatory compensation for memory decay. The model predicts systematic variation in norm-setting behaviour across agents and contexts. For instance, an agent targeting 90% test performance ($\rho^* = 0.9$) who expects 20% decay ($\delta_{\text{retention}} = 0.2$) must achieve 108% mastery during acquisition. Moreover, the framework anticipates differential standards across learning contexts: conceptual understanding tasks (G_1, with $\delta_{\text{retention}} = 0.1$) vs. verbatim recall tasks (G_2, with $\delta_{\text{retention}} = 0.2$) yield distinct acquisition targets (99% vs. 108% respectively) even under identical performance goals.

M-G-V (Learning) Cycle For learning cycles $\tau \in \{1, \ldots, T_{\text{learn}}\}$, let $\mathcal{T}_\tau = \{i_j : j \in \mathcal{J}_\tau\}$ denote the set of items remaining in the task at cycle τ, where $\mathcal{J}_\tau \subseteq \{1, 2, \ldots, N\}$ represents the indices of items still requiring learning. Φ_τ^{STM} represent the cumulative learning experience in working memory.

WHILE $\mathcal{S}_\tau = \text{ACTIVE}$:

1. **MONITOR**: *Assess current mastery for each item $i_j \in \mathcal{T}_\tau$.*

   Monitoring involves retrieving metacognitive knowledge and generating metacognitive experiences about the current learning state.

$$\mathcal{MK}_\tau^{\text{STM}} = \mathcal{MK}_{\tau-1}^{\text{STM}} \cup \text{retrieve}_\theta(\mathcal{MK}, \mathcal{ME}_{\tau-1}) \quad \text{if } \tau > 0$$
$$\mathcal{ME}_{\tau,j} = \begin{cases} [\text{EOL}_{\tau,j}, \text{null}] & \text{if } \tau = 0 \\ [\text{FOK}_{\tau,j}, \text{null}] & \text{if } \tau > 0 \end{cases}$$

   where:

$$\text{EOL}_{\tau,j} = \text{feel}(i_j) \oplus \text{assess}(i_j, \mathcal{MK}_\tau^{\text{STM}}) \qquad \text{if } \tau = 0$$
$$\text{FOK}_{\tau,j} = \text{feel}(i_j, \mathcal{CO}_{\tau-1,j}) \oplus \text{assess}(i_j, \mathcal{CO}_{\tau-1,j}, \mathcal{MK}_\tau^{\text{STM}}) \qquad \text{if } \tau > 0$$

   Metacognitive experiences are represented as vectors, reflecting Nelson and Narens' proposal of their multidimensional nature. Both Ease of Learning (EOL) and Feeling of Knowing (FOK) are immediate phenomenological experiences that emerge during cognitive tasks, illustrating how subjective feelings support monitoring functions, serving as the primary input for subsequent control decisions [Nelson and Narens, 1990, p. 160].

2. **GENERATE**: *Transforms monitoring outputs into executable learning actions.*

   Resources are allocated inversely proportional to their EOL or FOK, and strategy selection integrates metacognitive inputs to map learning methods to individual items.

$$r_{\tau,j} = R_{\text{total}} \times \frac{w_j}{\sum_{k=1}^{N} w_k}, \quad \text{where } w_j = (\mathcal{ME}_{\tau,j}[1])^{-1}$$
$$\sigma_{\tau,j} = \text{select}(s \in \mathcal{MK}_{\text{Strategy}} \mid i_j, r_{\tau,j}, \mathcal{ME}_{\tau,j}, \mathcal{MK}_\tau)$$

   The learning plan $\mathcal{P}_{\tau,j} = (i_j, r_{\tau,j}, \sigma_{\tau,j})$ is executed to produce cognitive outcomes (new memory state).

$$\mathcal{CO}_{\tau,j} = \text{execute}(\mathcal{P}_{\tau,j})$$

3. **VERIFY**: *Assess learning progress and determines cycle continuation.*

   Judgements of Learning (JOL) evaluate current mastery levels following cognitive outcomes.

$$\text{JOL}_{\tau,j} = \text{feel}(i_j, \mathcal{CO}_{\tau,j}) \oplus \text{assess}(i_j, \mathcal{CO}_{\tau,j}, \mathcal{MK}_\tau)$$
$$\mathcal{ME}_{\tau,j} = [\mathcal{ME}_{\tau,j}[1], \text{JOL}_{\tau,j}]$$
$$\Phi_{\tau,j}^{\text{STM}} = (\mathcal{ME}_{\tau,j}, i_j, r_{\tau,j}, \sigma_{\tau,j}, \mathcal{CO}_{\tau,j})$$
$$\Phi_\tau^{\text{STM}} = \Phi_{\tau-1}^{\text{STM}} \cup \{\Phi_{\tau,j}^{\text{STM}} : j \in \mathcal{J}_\tau\}$$

For each item $i_j \in \mathcal{T}_\tau$, an agent computes the mastery discrepancy. Items that have reached the norm of study are removed from further consideration. Thus, $\mathcal{T}_{\tau+1} = \{i_j : j \in \mathcal{J}_{\tau+1}\}$ contains only items still requiring learning.

$$\Delta_{\tau,j} = \mathcal{N}_s - \text{JOL}_{\tau,j}$$
$$\mathcal{J}_{\tau+1} = \{j \in \mathcal{J}_\tau : \Delta_{\tau,j} > 0\}$$

Learning continues as long as any item remains below the mastery threshold:

$$\mathcal{S}_{\tau+1} = \begin{cases} \text{ACTIVE} & \text{if } \mathcal{J}_{\tau+1} \neq \emptyset \\ \text{TERMINATE} & \text{otherwise} \end{cases}$$

B.2 Retrieval Process

Initialisation Given a retrieval query $\mathcal{Q}$, the agent establishes retrieval goals and accesses contextually relevant metacognitive knowledge for search control.

$$\mathcal{MK}_0^{\text{STM}} = \text{retrieve}_\theta(\mathcal{MK}, \mathcal{Q})$$

Nelson and Narens [1990] conceptualise FOK through the dual-counter hypothesis: one component accumulates evidence for information presence in memory (affirmative FOK, FOK_τ^+), while the other accumulates evidence for information absence, consistent with 'knowing not' [Kolers and Palef, 1976] (negative FOK, FOK_τ^-). This dual mechanism enables both continued search when positive evidence accumulates and efficient termination when negative evidence dominates, preventing exhaustive search behaviour.

The initial thresholds $\lambda_{\text{confidence}}^{(0)}$ and $\lambda_{\text{FOK}}^{(0)}$ are established through the agent's privileged access to personal metacognitive calibration history within $\mathcal{MK}_0^{\text{STM}}$:

$$\lambda_{\text{FOK}}^{(0)} = \text{median}(\{||\text{FOK}|| : \text{successful retrievals in } \mathcal{MK}_0^{\text{STM}}\})$$
$$\lambda_{\text{confidence}}^{(0)} = \text{median}(\{\text{confidence}_\tau : \text{correct outputs in } \mathcal{MK}_0^{\text{STM}}\})$$

FOK thresholds are calibrated based on successful retrievals – episodes where dual-counter FOK assessment correctly predicted retrieval outcomes, with $||\text{FOK}_\tau||$ (L1 norm) capturing the magnitude of metacognitive evidence. Confidence thresholds follow analogous calibration, reflecting historical accuracy at different confidence levels. This personalised approach embodies the No-Magic Hypothesis by utilising recallable metacognitive knowledge whilst accommodating domain-specific variations in metamemory accuracy.

M-G-V (Search) Process For search cycles $\tau \in \{0, 1, \ldots, T_{\text{search}}\}$, let $\mathcal{A}_\tau$ represent the current answer state (retrieved answer or null), and Ω_τ^{STM} represent the cumulative retrieval experience in working memory.

WHILST search is active:

1. **MONITOR**: *Assess Feeling-of-Knowing (FOK) and retrieval accessibility.*
   The metacognitive decision to initiate search relies on rapid, preliminary FOK judgement that operates faster than actual recall, enabling efficient search control [Reder, 1987, 1988]. Following the No-Magic Hypothesis, FOK monitoring accesses recallable item attributes – acquisition history, partial cues, contextual associations – rather than directly tapping unconscious memory states.

$$\mathcal{MK}_\tau^{\text{STM}} = \begin{cases} \text{retrieve}_\theta(\mathcal{MK}, \mathcal{Q}) & \text{if } \tau = 0 \\ \mathcal{MK}_{\tau-1}^{\text{STM}} \cup \text{retrieve}_\theta(\mathcal{MK}, \text{FOK}_{\tau-1}) & \text{if } \tau > 0 \end{cases}$$
$$\text{FOK}_\tau = \begin{bmatrix} \text{FOK}_\tau^+ \\ \text{FOK}_\tau^- \end{bmatrix} = \begin{cases} \text{feel}(\mathcal{Q}) \oplus \text{assess}(\mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}}) & \text{if } \tau = 0 \\ \text{feel}(\mathcal{Q}, \mathcal{A}_{\tau-1}) \oplus \text{assess}(\mathcal{Q}, \mathcal{A}_{\tau-1}, \mathcal{MK}_\tau^{\text{STM}}) & \text{if } \tau > 0 \end{cases}$$

At $\tau = 0$, preliminary FOK assessment determines search initiation through rapid accessibility evaluation using the dual-counter system. For subsequent cycles ($\tau > 0$), ongoing FOK monitoring incorporates previous search outcomes ($\mathcal{A}_{\tau-1}$) to reassess continued retrieval likelihood, with both affirmative (FOK_τ^+) and negative (FOK_τ^-) counters updating based on accumulating evidence.

2. **GENERATE**: *Deliberately attend to search cues and execute automatic search.*

Following Nelson and Narens' insight that search execution is automatic once initiated, the generation phase focuses on conscious cue attention rather than strategy selection. The dual-counter FOK hypothesis provides metacognitive control over cue generation intensity, reflecting the principle that monitoring should adaptively influence control processes.

$$\mathcal{S}_\tau = \begin{cases} \text{ACTIVE}_{\text{intensive}} & \text{if } ||\text{FOK}_\tau|| < \lambda_{\text{FOK}}^{(\tau)} \\ \text{ACTIVE}_{\text{standard}} & \text{if } ||\text{FOK}_\tau|| \geq \lambda_{\text{FOK}}^{(\tau)} \wedge \text{FOK}_\tau^+ > \text{FOK}_\tau^- \\ \text{TERMINATE} & \text{if } ||\text{FOK}_\tau|| \geq \lambda_{\text{FOK}}^{(\tau)} \wedge \text{FOK}_\tau^- > \text{FOK}_\tau^+ \end{cases}$$

The search intensity logic operates through evidence-based decision making. When the total magnitude of metacognitive evidence falls below the threshold ($||\text{FOK}_\tau|| < \lambda_{\text{FOK}}^{(\tau)}$), insufficient evidence has accumulated from both counters to make a reliable continuation decision. This triggers intensive cue attention to gather additional metacognitive information, preventing premature termination based on weak or ambiguous signals. When sufficient evidence exists ($||\text{FOK}_\tau|| \geq \lambda_{\text{FOK}}^{(\tau)}$), the system evaluates counter dominance: positive dominance ($\text{FOK}_\tau^+ > \text{FOK}_\tau^-$) indicates sufficient evidence for item presence to warrant continued search with standard attention, while negative dominance ($\text{FOK}_\tau^- > \text{FOK}_\tau^+$) provides sufficient evidence for item absence to justify search termination.

If $\mathcal{S}_\tau = \text{ACTIVE}$, the agent deliberately attends to retrieval cues that trigger automatic pattern-recognition-guided search, with attention determined by metacognitive confidence.

$$\text{cue}_\tau = \begin{cases} \text{attend}_{\text{intensive}}(\mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}}) & \text{if } \mathcal{S}_\tau = \text{ACTIVE}_{\text{intensive}} \\ \text{attend}_{\text{standard}}(\mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}}) & \text{if } \mathcal{S}_\tau = \text{ACTIVE}_{\text{standard}} \end{cases}$$

Once cues are consciously attended to, the search process $\text{search}_{\text{auto}}(\cdot)$ operates automatically through pattern recognition. Due to this automatic nature, $\mathcal{A}_\tau$ for consecutive cycles $\tau = 0, \ldots, k$ may yield identical results, reflecting the deterministic nature of automatic search.

$$\mathcal{A}_\tau = \text{search}_{\text{auto}}(\text{cue}_\tau)$$

3. **VERIFY**: *Evaluate retrieved answers based on confidence, update thresholds, and determine continuation.*

According to Nelson and Narens [1990], confidence governs output decisions for retrieved answers, while FOK governs continuation decisions when no answer is found, with both involving dynamic thresholds that can change during search.

$$\text{confidence}_\tau = \begin{cases} \text{assess}(\mathcal{A}_\tau, \mathcal{Q}, \mathcal{MK}_\tau^{\text{STM}}) & \text{if } \mathcal{A}_\tau \neq \text{null} \\ 0 & \text{if } \mathcal{A}_\tau = \text{null} \end{cases}$$

$$\text{decision}_\tau = \begin{cases} \text{OUTPUT } \mathcal{A}_\tau & \text{if } \mathcal{A}_\tau \neq \text{null} \wedge \text{confidence}_\tau \geq \lambda_{\text{confidence}}^{(\tau)} \\ \text{CONTINUE} & \text{if } \mathcal{A}_\tau \neq \text{null} \wedge \text{confidence}_\tau < \lambda_{\text{confidence}}^{(\tau)} \\ \text{CONTINUE} & \text{if } \mathcal{A}_\tau = \text{null} \wedge \text{FOK}_\tau^+ > \text{FOK}_\tau^- \\ \text{OUTPUT null (omission)} & \text{if } \mathcal{A}_\tau = \text{null} \wedge \text{FOK}_\tau^- > \text{FOK}_\tau^+ \end{cases}$$

This decision structure distinguishes between two primary error pathways identified by Nelson and Narens: (1) *Commission errors* occurring when $\mathcal{A}_\tau \neq \text{null}$ but the outputted answer is incorrect, typically associated with high confidence but incorrect retrieval; and (2) *Omission errors* occurring when search terminates without producing an answer ($\mathcal{A}_\tau = \text{null}$), often following prolonged search with declining FOK.

The retrieval experience accumulates in working memory, creating a comprehensive search history that informs adaptive threshold adjustment:

$$\Omega_{\tau}^{\text{STM}} = \begin{cases} [(\text{FOK}_{\tau}, \text{cue}_{\tau}, \mathcal{A}_{\tau}, \text{confidence}_{\tau})] & \text{if } \tau = 0 \\ \Omega_{\tau-1}^{\text{STM}} \cup [(\text{FOK}_{\tau}, \text{cue}_{\tau}, \mathcal{A}_{\tau}, \text{confidence}_{\tau})] & \text{if } \tau > 0 \end{cases}$$

Following the principle of satisficing [Simon, 1979], both confidence and FOK thresholds undergo dynamic adjustment based on accumulated search burden. This reflects the psychological tendency for acceptance criteria to progressively lower as the cost of continued searching increases. The satisficing adjustment factor captures this adaptive mechanism:

$$\beta_{\tau} = \exp(-\alpha \cdot (\tau + \sum_{(\mathcal{A}_i, \text{conf}_i) \in \Omega_{\tau}^{\text{STM}}} \mathbf{1}[\mathcal{A}_i = \text{null} \vee \text{confidence}_i < \lambda_{\text{confidence}}^{(i)}]))$$

where α represents the satisficing adjustment rate, and the exponential decay function models the psychological burden accumulating from both temporal persistence (τ) and retrieval failures (unsuccessful attempts or low-confidence outcomes). This burden manifests as decreasing acceptance standards, operationalised through threshold reduction:

$$\lambda_{\text{confidence}}^{(\tau+1)} = \lambda_{\text{confidence}}^{(\tau)} \cdot \beta_{\tau}$$
$$\lambda_{\text{FOK}}^{(\tau+1)} = \lambda_{\text{FOK}}^{(\tau)} \cdot \beta_{\tau}$$

This adaptive mechanism ensures that answers previously deemed inadequate may become acceptable as search costs accumulate. Consequently, at cycle $\tau + 1$, a previously retrieved answer might satisfy the lowered confidence threshold and be output, even though it failed to meet the more stringent earlier criteria.

The search state for the next cycle is determined by:

$$\mathcal{S}_{\tau+1} = \begin{cases} \text{ACTIVE} & \text{if decision}_{\tau} = \text{CONTINUE} \\ \text{TERMINATE} & \text{if decision}_{\tau} \in \{\text{OUTPUT } \mathcal{A}_{\tau}, \text{OUTPUT null}\} \end{cases}$$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state this is a theoretical contribution that formalises metacognitive theories, not an empirical study claiming performance improvements.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper explicitly states it makes no empirical claims and that translating psychological concepts to computational systems remains speculative.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper formalises existing psychological theories into computational frameworks rather than proving new theoretical results.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results?

   Answer: [NA]

   Justification: The paper does not include experiments; it presents theoretical formalisations of cognitive science theories.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code?

   Answer: [NA]

   Justification: The paper is purely theoretical and does not involve experiments requiring code or data.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details?

   Answer: [NA]

   Justification: The paper does not include experiments.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined?

   Answer: [NA]

   Justification: The paper does not include experiments.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources?

   Answer: [NA]

   Justification: The paper does not include experiments.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform with the NeurIPS Code of Ethics?

Answer: [Yes]

Justification: The theoretical work conforms with ethical guidelines and poses no ethical concerns.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts?

    Answer: [NA]

    Justification: This is foundational theoretical work without direct societal applications or deployment considerations.

11. **Safeguards**

    Question: Does the paper describe safeguards for responsible release?

    Answer: [NA]

    Justification: The paper poses no risks as it neither releases models nor datasets, only theoretical formalisations.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets properly credited?

    Answer: [NA]

    Justification: The paper does not use existing code, data, or model assets.

13. **New assets**

    Question: Are new assets introduced in the paper well documented?

    Answer: [NA]

    Justification: The paper does not release new assets.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include full details?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

15. **Institutional review board (IRB) approvals**

    Question: Does the paper describe potential risks incurred by study participants?

    Answer: [NA]

    Justification: The paper does not involve research with human subjects.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important component of the core methods?

    Answer: [NA]

    Justification: LLMs are not used as a component of the core theoretical formalisations presented.