

Beyond Prediction: A Cognitive Architecture for AI Systems Capable of Reasoning, Planning, Learning, and Autonomous Decision-Making

Naseef Talukder

Faculty of ITHM Department, Primeasia University, Dhaka

Ridwan Haque

Faculty of business and management, UCSI university

Abstract: Contemporary AI systems exhibit strong predictive and generative capabilities, yet autonomous performance in dynamic environments also requires integrated reasoning, planning, learning, and action. This paper proposes a cognitive architecture that organizes these capabilities into a closed loop: perceptual inputs inform structured representations and memory; reasoning and goal management guide hierarchical planning; action outcomes update the system's knowledge and future decisions. The architecture combines neural methods for learning from data with symbolic representations for structured inference and interpretable planning. It further treats uncertainty estimation and metacognitive monitoring as mechanisms for adapting behavior when knowledge or environmental conditions are incomplete. By connecting continual adaptation with explicit planning and evaluation, the proposed framework offers a conceptual basis for AI systems designed to act reliably beyond prediction alone.

Keywords: *Cognitive Architectures; Autonomous AI; Neuro-Symbolic Reasoning; Hierarchical Planning; Continual Learning; Metacognition.*

I. INTRODUCTION

Artificial intelligence has advanced through data-driven learning, yet autonomous systems must do more than produce predictions: they must coordinate perception, reasoning, planning, and action in pursuit of complex goals. Planning and acting require deliberation that links goals to executable behavior while accommodating changing conditions. Cognitive architectures provide one approach to organizing such capabilities within integrated systems.

Despite this history, integrating learning, structured reasoning, and adaptive behavior remains a central challenge for cognitive systems. Neural methods support data-driven learning, whereas symbolic approaches facilitate explicit representation and inference; combining these strengths is a defining objective of neuro-symbolic AI. Systems operating in dynamic environments must also connect planning with learning and revise behavior in response to action outcomes.

This paper proposes a cognitive architecture that treats autonomous performance as a closed loop connecting perception, structured representations and memory, reasoning, goal management, hierarchical planning, and action. It combines neural learning with symbolic inference and planning, while incorporating uncertainty estimation and metacognitive monitoring to support adaptation when knowledge or environmental conditions are incomplete. This scope builds on work integrating cognitive architectures, knowledge, and foundation models for complex decision-making, and on research examining metacognition as a basis for adapting to unknown situations.

Why Correlation-Driven AI Is Insufficient for Complex, High-Stakes Decisions

Modern predictive systems estimate associations between inputs and outputs, but complex decisions require more than accurate classification. High-stakes agents must represent causes, preserve relevant history, compare possible actions, explain constraints, and revise decisions when conditions change. Opaque correlation-driven predictions can therefore leave experts without an account of why a recommendation was produced or how it should guide intervention, a problem identified in health-care decision support. Neural systems also remain vulnerable to unstable outputs and adversarially induced errors, while knowledge graphs often store facts without supporting adequate inference. The central limitation is functional: prediction estimates what may occur, whereas autonomy requires selecting and executing actions toward stated goals under uncertainty.

Defining Cognitive Autonomy Through Reasoning, Planning, Learning, and Goal-Directed Action Cognitive autonomy denotes an agent's capacity to maintain internal state, interpret observations, formulate or accept goals, reason over alternatives, plan actions, monitor consequences, and learn from outcomes. This definition follows the characterization of cognitive systems as adaptive, anticipatory, and purposive systems [1]. Reasoning supplies structured comparison; planning organizes actions over time; learning changes representations or policies; and goal-directed action connects internal computation to environmental consequences. Autonomy therefore concerns an integrated control process rather than a model's isolated language or prediction score. It also requires bounded self-direction, because an agent that generates its own actions without policy constraints can increase operational risk [2].

Review Scope, Evidence Clusters, and Comparative Evaluation Criteria

This review examines cognitive architectures, neuro-symbolic interfaces, planning, continual learning, memory, metacognition, embodiment, and evaluation. The evidence includes broad surveys, architecture proposals, primary robotic demonstrations, continual-learning experiments, planning studies, and recent agent-evaluation reviews. The comparison uses six criteria: representational adequacy, reasoning traceability, planning competence, adaptation across tasks, closed-loop robustness, and governance. These criteria extend evaluation beyond predictive accuracy toward system-level autonomy, consistent with calls for architectural criteria covering representation, organization, performance, and learning [3]. The next section establishes the architectural substrate on which these capacities depend.

Cognitive Architectures as the Structural Foundation of General Intelligent Agents Core Cognitive Abilities: Perception, Memory, Representation, Goal Management, and Action

A cognitive architecture specifies persistent structures and processes that coordinate perception, memory, representation, learning, reasoning, metareasoning, and action selection. A major survey identified 84 architectures, 49 of them actively developed, and catalogued more than 900 practical projects, demonstrating both the breadth of the field and the absence of a single accepted design [4]. Perception converts signals into usable states, while memory preserves task knowledge, episodes, procedures, and environmental regularities. Representation determines whether an agent can express objects, relations, causal dependencies, uncertainty, and goals at appropriate levels of abstraction. Goal management allocates attention and computation among competing objectives, and action selection converts deliberation into executable behavior.[5]

These components cannot be treated as independent services. A perceptual error can produce an incorrect symbolic state; a defective memory retrieval can bias planning; and a planner may select an action whose assumptions no longer hold. ICARUS illustrates this integration by linking conceptual and skill memories to performance and learning in an in-city driving task [6]. RCS similarly organizes sensory processing, world modeling, value judgment, behavior generation, and knowledge databases across multiple hierarchical levels [7]. Such architectures imply that general intelligence depends less on any isolated module than on stable interfaces among modules.

From Soar, ACT-R, ICARUS, BDI, Subsumption, and CLARION to Integrated Agent Architectures

The historical comparison of Soar, ACT-R, ICARUS, BDI, subsumption, and CLARION shows that each emphasizes a different account of cognition. Symbolic production systems support explicit problem solving and rule-based control; ACT-R links cognitive operations to psychologically motivated memory mechanisms; ICARUS separates concepts from skills; BDI organizes behavior around beliefs, desires, and intentions; subsumption prioritizes layered reactive control; and CLARION combines implicit and explicit knowledge. A comparative survey found that no reviewed architecture achieved the full range of human intelligence, although recurring design principles emerged for representation, learning, planning, and problem solving [8]. The lesson is not that one historical system should be selected unchanged, but that integration must preserve the distinct computational benefits of these traditions.[9], [10]

Architectures also differ in the location of control. BDI systems can respond efficiently through trigger-based plan libraries, yet fixed libraries constrain adaptation when unforeseen situations arise. Declarative planning integrated into a BDI interpreter permits goals to be specified independently of every contingency, increasing the agent's capacity to construct plans at runtime [11]. ICARUS contributes grounded conceptual and procedural memory, whereas subsumption contributes rapid responses with limited deliberation. A contemporary architecture should therefore treat these systems as design patterns within a larger control organization rather than as mutually exclusive alternatives.[12]

Reactive, Deliberative, Hierarchical, and Multi-Agent Control Structures

Reactive control maps current observations to rapid responses and is appropriate when deadlines are tight or the relevant policy is stable. Deliberative control constructs and evaluates action sequences, which supports long-horizon tasks but introduces computational delay and dependence on accurate models. Hybrid control combines these modes: MERLIN2 contains a knowledge base and symbolic planner alongside reactive and emergent behavioral components, and it was evaluated in simulation and in a real-world luggage-carrying task [13]. Its architecture demonstrates how immediate behavior can coexist with planning and monitoring.[14]

Hierarchy provides a second organizing principle. RCS places computational agents at multiple resolutions, with lower levels producing goal-seeking reactions and higher levels supporting planning and decision-making [7]. IDEA assigns agents to different abstraction layers while retaining common model and planning semantics, allowing reactive and deliberative behavior within guaranteed reaction-time constraints [15]. Multi-agent structures extend this arrangement to distributed systems, where agents may coordinate through shared models, communication, or negotiated goals. The unresolved issue is consistency across levels: a fast controller must know when to interrupt a plan, while a deliberator must incorporate changes detected by lower-level processes. This problem leads directly to interfaces between learned perception and structured reasoning.[16]

Neuro-Symbolic Integration: Connecting Learned Perception with Structured Reasoning Complementary Strengths and Persistent Conflicts Between Neural and Symbolic Paradigms

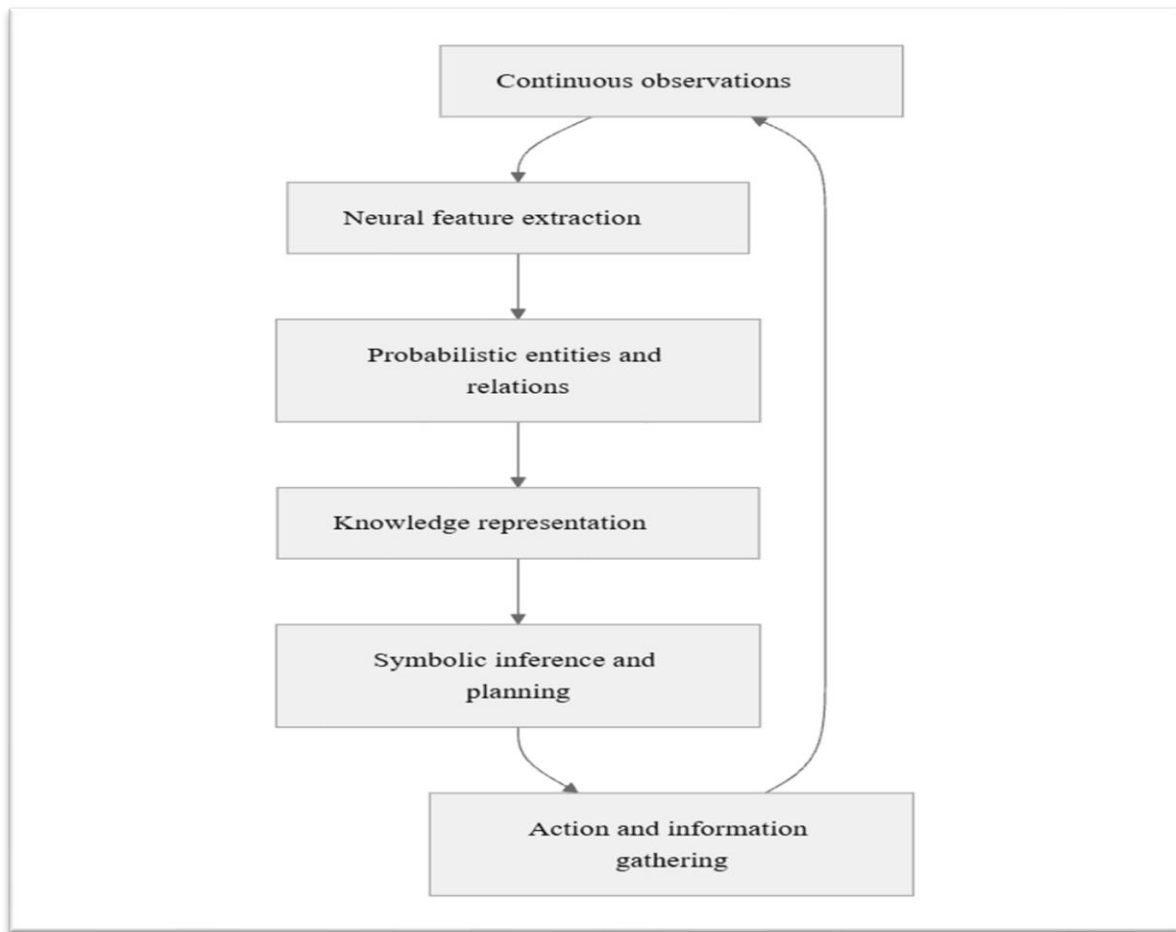
Neural computation handles noisy, high-dimensional input and can learn representations from data, whereas symbolic computation expresses relations, rules, constraints, and derivations. Reviews of neuro-symbolic AI describe this combination as a route toward systems that retain neural adaptability while adding logical inference and explanation [17], [18]. The distinction is functional rather than absolute: neural systems can perform limited relational operations, and symbolic systems can incorporate uncertainty, but their dominant strengths remain different. A cognitive agent needs both because perception rarely arrives in symbolic form, while long-horizon reasoning is poorly specified by local pattern similarity alone.

Integration nevertheless produces conflicts. Symbolic inference expects stable predicates and explicit semantics, whereas neural outputs are distributed, graded, and context-sensitive. End-to-end systems can reduce manual interfaces but face combinatorial growth in symbol grounding; sequential pipelines simplify verification but require extensive annotation and may propagate upstream errors [19]. Scalability is also unresolved when symbolic search expands over large knowledge bases or multimodal observations [17]. The architecture must therefore specify where uncertainty is represented, how symbols are grounded, and which component has authority when neural and symbolic conclusions disagree.

Interfaces Between Continuous Perception, Knowledge Representation, and Discrete Inference

A useful interface transforms sensor data into structured hypotheses rather than pretending that perception is certain. In one uncertainty-aware framework, a transformer and graph neural network produced probabilistic symbolic states from visual scenes, which an uncertainty-aware planner then used to select actions and gather information when confidence was low [20]. The reported tabletop system processed 10,047 simulated scenes, achieved an overall predicate F1 of 0.68, and reached success rates of 94%, 90%, and 88% on three stacking tasks, with a 90.7% average and planning within 15 milliseconds [20]. These results illustrate an interface in which confidence values remain attached to predicates and influence downstream planning.[21]

Knowledge representation must also support grounding, revision, and action semantics. Knowledge-processing methods for robots allow systems to reason about observations, action parameters, task structure, and the reasons for selected actions, but dynamic environments create persistent challenges for maintaining that knowledge [22]. Foundation models can reduce manual engineering by extracting symbolic features from raw data and generating programmatic interfaces for downstream answer-set reasoning [19]. Yet generated predicates require validation because linguistic or visual fluency does not guarantee correct reference to objects, relations, or operational constraints.



The interface is therefore a control boundary, not merely a translation layer. It should preserve provenance, confidence, temporal context, and the conditions under which a predicate was inferred. Without those attributes, symbolic reasoning can give a false appearance of certainty.

Foundation Models, Knowledge Graphs, Logic, and Explainable Autonomous Reasoning

Foundation models provide broad linguistic and multimodal priors that can assist perception, hypothesis generation, and interface construction. Their use in neuro-symbolic systems is attractive because manually encoding every concept and relation is costly, especially when observations are heterogeneous. NeSyGPT, for example, fine-tunes a vision-language foundation model to extract symbolic features and uses an answer-set program for a downstream task [19]. The approach addresses an engineering bottleneck, but it does not eliminate the need to test whether extracted symbols are complete, correctly grounded, and appropriate for the task. Knowledge graphs contribute persistent relational structure, while logic provides constraints and derivations that can be inspected. A hybrid proposal assigns generative networks the task of producing candidate solutions and symbolic inference the task of filtering them against verifiable rules, including negative knowledge about prohibited actions [23]. This division resembles a proposal-and-verification loop: neural generation expands the search space, and symbolic controls reject candidates that violate domain constraints. A convolutional-perception and Prolog-reasoning system for ARC reported an F1 score of 0.916 and classification accuracy of 0.93 while producing symbolic explanations [24]. These results are promising but remain task-specific; they do not establish reliable autonomy across changing environments.[25]

Explainability should consequently be defined at several levels. A system may expose a proof trace, the retrieved evidence, the plan that was selected, the uncertainty attached to a percept, or the policy that blocked an action. These forms answer different questions and should not be conflated with a natural-language summary generated after the decision. Cognitive neurosymbolic decision support links multimodal perception, symbolic controls, and expert knowledge to actionable recommendations in a mental-health support setting [26]. The evidence supports traceable decision support, but it does not justify treating explanations as proof of correctness.

Reasoning Under Perceptual Uncertainty and the Reliability–Interpretability Trade-off

Uncertainty must propagate from sensing through representation, inference, planning, and action. If a robot is uncertain whether an object is graspable, a planner may choose observation, defer execution, or select a robust alternative rather than treat the uncertain predicate as true. The uncertainty-aware stacking study directly links calibrated perceptual confidence to planning

convergence and reports higher task success than its strongest POMDP baseline by 10-14 percentage points [20]. This suggests that interpretability and reliability can reinforce one another when symbolic states retain probabilistic semantics.

The trade-off appears when richer models increase computation or when strict symbolic assumptions reject useful but imperfect predictions. Neural systems tolerate noise and incomplete specification, but their errors may be difficult to diagnose; symbolic systems expose constraints but can fail when predicates are missing or ontologies are too rigid [27]. Hybrid systems can also inherit both failure modes if an uncertain neural output is converted into an overconfident symbol. Reliable autonomy therefore requires calibrated confidence, abstention, contradiction handling, and action-level validation rather than a simple choice between neural flexibility and symbolic transparency. Planning provides the next test because it reveals whether representations support sustained action rather than isolated inference.[28]

Planning and Decision-Making Beyond Prediction Symbolic Planning, Hierarchical Planning, and Planning in Continuous Environments

Planning differs from prediction because it evaluates possible interventions under goals, constraints, and transition assumptions. Classical symbolic planners operate over states, actions, preconditions, and effects, making plans inspectable and compatible with domain knowledge. Planning research also includes motion, sensor-based, decision-theoretic, probabilistic, kinodynamic, and nonlinear forms, indicating that no single planner covers all physical and informational demands [29]. A cognitive architecture must therefore select planning representations according to the action horizon, uncertainty, and environmental dynamics.

Hierarchical planning reduces search by decomposing abstract objectives into subtasks and executable skills. High-level symbolic plans can guide lower-level controllers, while lower-level experience supplies information about whether abstract actions are feasible. An integration of AI planning with hierarchical reinforcement learning maps planning operators to options, using intrinsic rewards to maintain correspondence between symbolic transitions and learned behaviors [30]. This arrangement retains interpretable high-level structure while allowing robust learning of low-level actions.[31]

Continuous environments expose a grounding problem. A symbolic planner cannot operate effectively unless an agent knows which continuous states satisfy relevant predicates and how actions change their distributions. Konidaris and colleagues derived grounded abstract representations from high-level actions, showing that deterministic planning requires suitable sets of states, whereas probabilistic planning requires distributions over states [32]. Their work connected sensorimotor data to PDDL representations and demonstrated planning in a physical mobile-manipulation system [32]. The result supports a principled route from continuous experience to symbolic planning, although the method depends on identifiable actions and learnable abstractions.[31]

The Integration of Planning and Acting in Real-World Autonomous Systems

A plan has operational value only when execution monitors whether its assumptions remain valid. Automated planning and acting therefore treat deliberation and action as coordinated processes: the agent selects actions, observes consequences, updates its model, and replans when necessary [33]. This loop is especially important in robotics, where delays, occlusion, contact dynamics, and other agents can invalidate an otherwise correct plan. MERLIN2 integrates behavior generation, control, planning, monitoring, memory, learning, and explainability in a hybrid robotic architecture [13].

Real-time operation requires explicit allocation of computational effort. IDEA uses a common model and plan database across layers while supporting both reactive and deliberative planners under reaction-time requirements [15]. RCS distributes world modeling, value judgment, behavior generation, and knowledge across hierarchical agents, allowing local responses and higher-level deliberation [7]. These designs suggest that planning should be interruptible, bounded, and connected to a safety controller. A planner that cannot yield to urgent perception is unsuitable for physical autonomy, while a reactive controller without access to goals cannot manage extended tasks.

Learning Symbolic Skills and Abstract Representations for High-Level Plans

Learning symbolic skills addresses the gap between raw experience and reusable plans. The agent must infer which distinctions matter for action success, which effects persist, and which observations identify a relevant abstract state. Konidaris and colleagues formalized this problem for deterministic and probabilistic planning, showing how state sets or distributions can be learned from sensorimotor experience and expressed in a standard planning language [32]. Their physical robot demonstration indicates that abstract representations need not be hand-authored in every case.[34]

Language provides another route to skill abstraction. An integrated robot architecture combined multimodal categorization, reinforcement learning, grammar learning, and a Bayesian hidden Markov model so that the robot could associate concepts with actions and verbalize behavior [35]. The experiments support joint treatment of action understanding and language, but the abstract does not establish general transfer across users, tasks, or environments. Learning should therefore be evaluated by plan reuse, abstraction stability, recovery from failed execution, and the fidelity of learned predicates, not by task completion alone.[36]

The Planning Debate: Explicit Deliberation Versus Reactive and Learned Control

Explicit planning supplies compositionality, inspectable constraints, and the ability to reason about unobserved future states. Reactive and learned control supplies speed, tolerance to noise, and adaptation when a complete domain model is unavailable. The literature does not support a universal winner: AI planning is interpretable and efficient when logical specifications are available, whereas reinforcement learning is more tolerant of noise but is sample intensive and difficult to augment with external knowledge [30]. Fixed trigger-response plans are efficient but can fail under unforeseen conditions, while declarative planning increases runtime flexibility [11].

The strongest position is architectural coordination rather than replacement. High-level deliberation can set subgoals and constraints; learned skills can execute them; reactive layers can prevent immediate hazards; and monitoring can trigger replanning. Active predictive coding similarly proposes hierarchical world models that compose state dynamics and actions across abstraction levels for perception, action, and planning [37]. This arrangement shifts the question from whether planning is necessary to which decisions require deliberation, which can be cached as skills, and how the system detects when a familiar policy no longer applies. Continual learning determines whether those adaptations accumulate without destroying prior competence.[38]

Continual Learning, Memory, and Metacognition for Adaptive Agents Catastrophic Forgetting and the Challenge of Learning Tasks Across Time

Continual autonomy requires an agent to acquire new tasks while retaining prior capabilities. Sequential learning creates interference when new updates alter representations that support earlier tasks. In a comparison of humans and neural networks, standard supervised networks showed catastrophic forgetting under blocked task training, whereas an unsupervised generative model that learned a structured stimulus embedding reduced forgetting [39]. The authors relate human performance to factorized task representations that separate rules within a shared stimulus space [39]. This finding supports representation design as a central continual-learning issue, rather than treating forgetting as a problem solved only by replay.

The challenge is broader in autonomous agents because task distributions, goals, tools, and environments change simultaneously. A system may preserve a parameter configuration while losing the operational meaning of stored knowledge, or may retain a skill whose preconditions have changed. Reviews of continual task learning emphasize how neural task representations change in geometry and dimensionality as new tasks are acquired [40]. A cognitive architecture therefore needs mechanisms for task context, protected knowledge, revision, and selective reuse.

Hierarchical Memory, Prototype Replay, Knowledge Updating, and World-Model Maintenance

Memory should distinguish transient observations, episodes, procedures, semantic knowledge, and world-model beliefs. Hierarchical organization permits an agent to retrieve a concrete prior episode for immediate control while using generalized knowledge for planning. Prototype-guided memory replay addresses storage pressure by retaining synthetic prototypes to guide selection of replay samples, with reported gains in accuracy and efficiency on continual text-classification benchmarks [41]. The result is relevant to cognitive architectures because memory capacity, retrieval policy, and abstraction jointly determine what later learning can preserve.[38]

Replay alone cannot maintain an autonomous world model. The agent must identify stale facts, contradictory reports, changed affordances, and knowledge whose provenance is weak. A provenance-aware memory system can connect observations and conclusions to sources and outcomes, while a generative world model can simulate consequences at multiple temporal scales [42]. These architectural proposals remain more developed at the specification level than at broad deployment validation. Memory maintenance should therefore measure retention, update accuracy, retrieval relevance, contradiction resolution, and the effect of outdated beliefs on action.

Simultaneous Learning and Planning in Dynamic Environments

Separating planning from learning can create a mismatch: the planner assumes a fixed model while the learner changes action knowledge, or learning waits until planning has already failed. A simultaneous learning-and-planning formulation updates action plans and interaction models during operation in a partially observed environment [43]. Its hierarchical, object-oriented formulation removes the requirement for a fixed cognitive cycle that strictly separates planning and learning [43]. An unmanned-vehicle example illustrates the intended connection between dynamic planning and model acquisition, although the abstract does not establish broad empirical generality.

A closed-loop learner should treat plans as hypotheses about future transitions. Execution produces evidence about preconditions, effects, duration, and uncertainty; the system then revises the model or selects another strategy. This approach resembles hypothesis-driven cognitive cycles in which reasoning, action, evaluation, and learning remain connected [44]. It also aligns with architectures that combine intrinsic motivation, world modeling, hierarchical planning, and compute budgets [42]. The unresolved design question is how much exploration an agent may undertake before governance constraints require human approval.

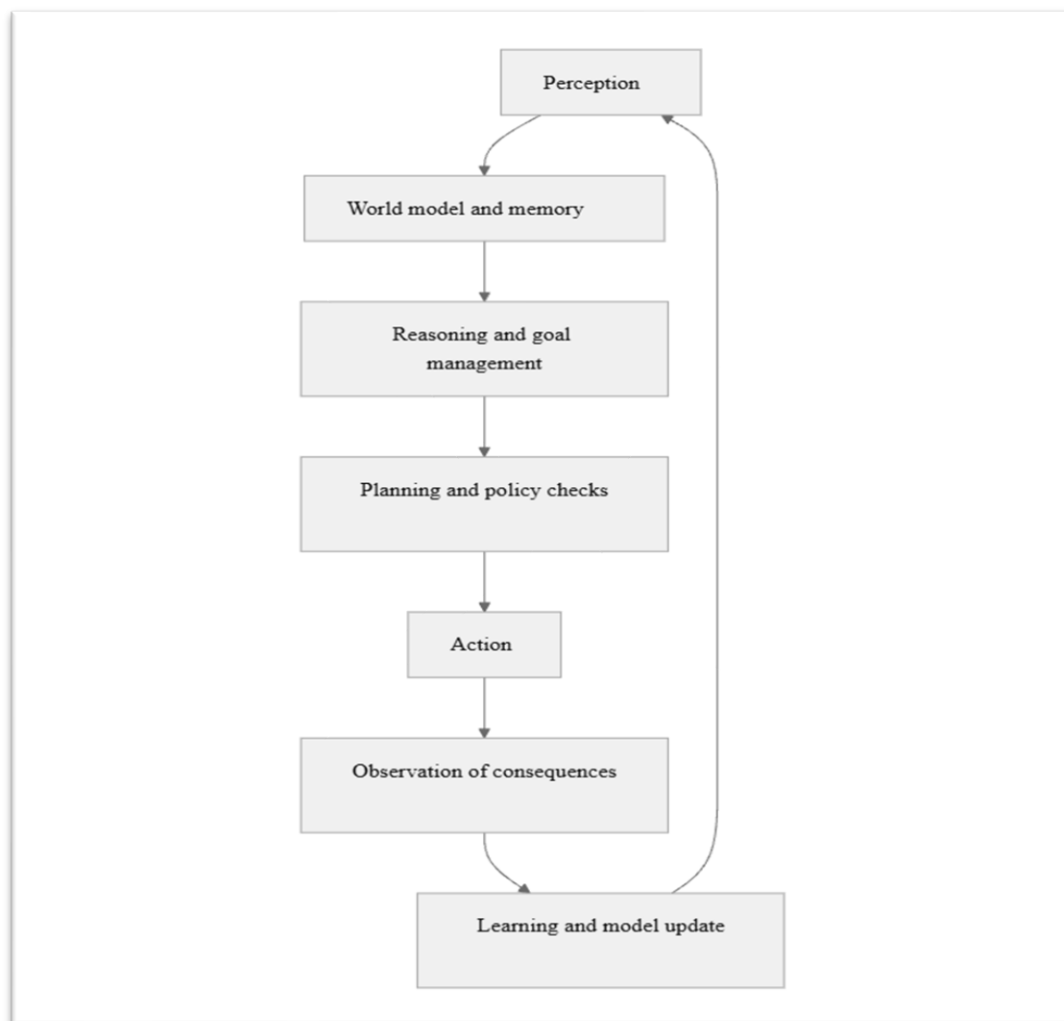
Metacognition, Self-Regulation, and Adaptation to Unknown Situations

Metacognition adds estimates about competence, uncertainty, strategy suitability, and resource use. MUSE operationalizes competence awareness and strategy selection through continual self-assessment, then uses those assessments to regulate behavior in unfamiliar environments [45]. Its implementations use either world modeling or large language models, and the reported agents improved self-regulation on out-of-distribution tasks relative to model-based reinforcement learning and prompt-based agent approaches [45]. The contribution is architectural: an agent can ask whether it knows how to proceed before committing to a strategy.

Self-regulation should include more than confidence reporting. The agent may seek information, reduce action scope, switch planners, request assistance, or stop when its model is unreliable. Governance-oriented architectures extend this idea through claim boundaries, approval gates, provenance, reversibility, and validation workflows [2]. These mechanisms convert metacognition into operational control, but self-assessment remains vulnerable to shared representation errors: an agent can be confidently wrong when perception, memory, and evaluation rely on the same defective model. Independent checks and outcome-based monitoring are therefore necessary before metacognitive claims can support unsupervised deployment.

Closed-Loop Cognitive Control for Autonomous Embodied and Agentic Systems Perception–Memory–Reasoning–Planning–Action–Learning as an Integrated Cognitive Cycle

An autonomous agent can be represented as a recurrent cycle rather than a linear pipeline. Perception updates a world model; memory supplies prior episodes and knowledge; reasoning identifies possible interpretations; planning selects an action sequence; execution changes the environment; and learning evaluates the result. This organization matches cognitive architectures that coordinate sensory processing, world modeling, value judgment, behavior generation, and knowledge across levels [7]. It also matches governed architectures that explicitly connect observation, memory formation, decision support, planning, execution, governance, and adaptation [2].



The cycle must preserve temporal continuity. Each decision should record the observations, retrieved knowledge, assumptions, selected plan, executed actions, and resulting evidence. Such records permit failure analysis and support selective learning rather than indiscriminate parameter updates. A system that generates an answer but cannot maintain state across actions is therefore less autonomous than one that performs narrower tasks with auditable memory and recovery.

Behavior Generation in Interactive Robots and Physical Agents

Physical agents expose errors that text-only evaluation can conceal. A robot must coordinate perception, localization, object relations, motor constraints, timing, social interaction, and recovery from contact or navigation failure. Cognitive-architecture reviews in autonomous robotics identify a recent movement toward hybrid systems that combine symbolic planners, neural networks, behavior trees, and robotic middleware [46]. The review selected 22 publications from an initial 502 studies, indicating both active research and a narrow evidence base for comparative conclusions [46].

MERLIN2 provides a concrete example by combining deliberative, reactive, and emergent components for behavior generation and monitoring, with evaluation in simulation and a physical luggage-carrying task [13]. Integrated action-and-language robotics similarly connects multimodal categorization, reinforcement learning, grammar learning, and robot behavior [35]. These systems demonstrate feasibility, but task-specific demonstrations do not establish robust autonomy in open-ended environments. Evaluation must include interruption, recovery, unanticipated objects, changing goals, and safe failure.

Language, Tool Use, Desktop Interaction, and Multi-Step Agent Execution

Language can function as an interface to goals, tools, explanations, and social coordination, but fluent generation does not guarantee reliable execution. Agentic systems extend language models through memory, planning, external tools, and environmental interaction [47]. In web environments, Agent Q combines guided Monte Carlo tree search, self-critique, and iterative preference optimization over successful and unsuccessful trajectories; its reported booking success rose from 18.6% in a zero-shot setting to 81.7% after one day of data collection and to 95.4% with online search [48]. These figures concern particular booking scenarios and should not be generalized to autonomous competence across domains.

Desktop and tool agents require action validation because a mistaken click, command, or transaction can have durable consequences. A governed desktop architecture separates cognition from language generation and emphasizes provenance, reversibility, approval boundaries, policy enforcement, and auditability [2]. This separation allows language to propose actions without granting it unrestricted execution authority. Multi-step performance should therefore be assessed through trajectory quality, tool correctness, recovery, cost, and policy compliance, rather than final textual success alone.[49]

Governance, Self-Monitoring, and Decision Oversight Within Autonomous Architectures

Governance must operate inside the action loop. It should constrain goals, inspect plans, verify evidence, enforce permissions, monitor execution, and determine when a human must approve or review an action. The proposed four-tier architecture places cognitive processing, metacognitive monitoring, executive decision-making, and guardrails in separate functional tiers [50]. A governed agent platform similarly emphasizes bounded autonomy, claim boundaries, validation, and explicit approval points [2]. These proposals converge on oversight as an architectural function rather than an external afterthought.[51]

Self-monitoring should detect uncertainty, resource exhaustion, policy conflict, and divergence between expected and observed outcomes. The agent can then pause, request information, select a reversible action, or transfer control. Such mechanisms reduce the consequences of model error but do not prove that the system is safe; governance rules can be incomplete, misconfigured, or bypassed by tool interfaces. Evidence synthesis must therefore evaluate both cognitive competence and the conditions under which competence fails.[52]

Evidence Synthesis and Evaluation: Measuring Cognitive Competence Rather Than Prediction Accuracy Comparative Evaluation Dimensions: Reasoning, Planning, Adaptation, Robustness, Explainability, and Autonomy

Evaluation should treat cognitive competence as a vector of related capacities. Reasoning concerns valid inference and evidence use; planning concerns goal-directed sequencing; adaptation concerns learning under changing tasks; robustness concerns uncertainty and perturbation; explainability concerns traceable grounds for decisions; and autonomy concerns sustained action with bounded oversight. Cognitive-architecture research already evaluates perception, attention, action selection, memory, learning, reasoning, and metareasoning as distinct abilities [4]. Agentic-AI evaluation surveys likewise organize assessment around performance, reliability, governance, reasoning, planning, and collaboration [53].

A single success score conceals trade-offs among these dimensions. An agent may complete a task through a brittle shortcut, produce an accurate answer without an adequate explanation, or achieve high reward by violating safety constraints. Evaluation should therefore report task outcome alongside trajectory validity, plan revisions, uncertainty calibration, resource use, policy compliance, and human intervention. The evidence base supports this multidimensional orientation, but common protocols remain inconsistent across robotics, software, language, and multi-agent settings.[54]

Evidence cluster	Primary strength	Main limitation	Evaluation implication
Cognitive-architecture surveys	Broad coverage of abilities and applications [4]	Heterogeneous tasks and measures	Compare component coverage before performance scores
Neuro-symbolic robotic planning	Grounded uncertainty and action success [20]	Focused tabletop domain	Test transfer and calibration under new observations
Continual-learning experiments	Retention and interference mechanisms [39]	Controlled task sequences	Add long-duration, multi-domain evaluation
Architecture demonstrations	Integrated behavior and monitoring [13]	Limited task breadth	Measure recovery, interruption, and deployment constraints
Agent benchmarks and reviews	Cross-domain task taxonomy [55]	Binary and task-completion emphasis	Add safety, cost, maintainability, and workflow metrics

Benchmark Performance Versus Deployment Viability in Agentic AI

Benchmark performance is an incomplete proxy for deployment viability. A recent review of 15 agent benchmarks reports that 13 of 15 rely exclusively on binary success measures, while none incorporates safety or security and none includes cost efficiency in its primary scoring protocol [55]. Such metrics can reward completion even when an agent uses excessive resources, produces insecure code, violates permissions, or leaves an unrecoverable state. They also provide limited information about planning coherence and the sequence of errors preceding success.

Deployment evaluation must examine environmental validity, repeatability, maintenance, integration, and human workflow. An agent that succeeds in a static benchmark may fail when interfaces change, tools return ambiguous results, or a user revises the goal midway through execution. Robotic reviews similarly identify variation in architectures, tools, middleware, and metrics, which complicates direct comparison [46]. Benchmark suites should therefore include perturbation tests, longitudinal adaptation, held-out tools, intervention costs, and explicit failure reporting.

Conflicting Evaluation Priorities: Capability Gains, Formal Guarantees, Safety, and Computational Cost

Cognitive systems face competing objectives. More expressive reasoning can increase search cost; richer memory can increase retrieval latency; end-to-end training can improve adaptation while weakening formal guarantees; and strict symbolic constraints can reduce unsafe actions while rejecting useful behavior. Neuro-symbolic surveys identify scalability and interpretability as persistent tensions [17], while planning and reinforcement-learning research contrasts the efficiency and interpretability of symbolic planning with the noise tolerance and learning flexibility of reinforcement learning [30]. Evaluation must report these trade-offs rather than rank systems using accuracy alone.

Formal guarantees also have limited scope. A planner can guarantee properties relative to a model, but those guarantees weaken when perception is uncertain, the world changes, or the action interface behaves unexpectedly. The uncertainty-aware planning framework addresses this problem by carrying confidence from perception into symbolic decisions and by gathering information when confidence is low [20]. Its results support the value of calibrated uncertainty, yet the reported task remains a constrained robotic domain. Capability claims should consequently state the environment, model assumptions, computation, and oversight conditions under which they hold.

Planned Evidence Displays: Architecture Comparison Table, Capability Radar Graph, and Closed-Loop System Diagram

The review should present an architecture comparison table that records modules, representation types, learning mechanisms, planning modes, embodiment, monitoring, and evidence maturity. A capability radar graph can display normalized scores only after a common protocol defines how reasoning, adaptation, robustness, explainability, and autonomy are measured. It should not convert incomparable task results into artificial rankings. The closed-loop system diagram should then connect perception, memory, reasoning, planning, action, learning, and governance, making explicit where evidence supports a component and where the literature remains preliminary.

These displays have different purposes. The table preserves methodological detail, the radar graph summarizes comparable measurements, and the diagram exposes dependencies and control boundaries. Together they prevent a high score in one task from being mistaken for general cognitive autonomy. The evidence synthesis therefore ends with a methodological requirement: future systems must be evaluated as governed, adaptive agents operating over time, not as predictors tested only on isolated outputs.[56]

A Unified Cognitive Architecture for Reasoning, Planning, Learning, and Autonomous Decision-Making Architectural Modules: Perception, Global Workspace, World Model, Memory, Reasoning, Planning, and Governance

A unified architecture should separate functions while preserving recurrent communication among them. **Perception** converts multimodal signals into uncertain state estimates; a **global workspace** prioritizes information for current deliberation; a **world**

model represents entities, relations, affordances, and anticipated consequences. Hierarchical architectures already combine sensory processing, world modelling, value judgment, behaviour generation, and knowledge storage across levels [7]. **Memory** should distinguish episodic experience, semantic knowledge, procedural skills, and provenance, while reasoning operates over grounded symbolic and probabilistic representations. Planning converts goals into executable actions, and **governance** constrains those actions through permissions, monitoring, audit trails, and approval boundaries [2].

Information Flow Across Neural, Symbolic, Hierarchical, and Metacognitive Components

Information should move bidirectionally rather than through a one-way prediction pipeline. Neural modules extract features and candidate predicates; symbolic modules test consistency, infer consequences, and construct plans; hierarchical controllers translate abstract plans into learned skills. Uncertainty must accompany perceptual representations into planning, allowing the agent to gather information when confidence is inadequate [20]. Metacognitive monitoring then estimates competence, selects strategies, and revises behaviour when task conditions exceed the agent's learned competence [45]. This organization links continuous sensing to discrete deliberation without treating either representation as sufficient on its own.

Design Principles for Compositionality, Continual Adaptation, Explainability, and Fault Tolerance Compositionality requires reusable representations of objects, actions, subgoals, and relations. Learned symbolic abstractions can connect sensorimotor data to planning operators, with probabilistic representations required when state transitions are uncertain [32]. Continual adaptation requires protected memory, replay, task separation, and world-model revision because sequential training can produce catastrophic forgetting [39], [41]. Explainability should expose perceptual evidence, retrieved knowledge, inference steps, plan alternatives, and authorization decisions. Fault tolerance should isolate uncertain modules, permit graceful degradation, and prevent unverified outputs from directly controlling high-consequence actions

Proposed Reference Architecture Diagram and Capability-to-Evidence Mapping Table

The reference cycle is shown below. It treats learning as a consequence of action and evaluation, rather than as an offline stage detached from deployment [44].



Capability	Architectural evidence
Grounded planning	Learned abstract representations support symbolic planning from sensorimotor data [32]
Hybrid control	MERLIN2 combines reactive, deliberative, and emergent behaviour systems [13]
Continual learning	Prototype replay and structured task representations address forgetting [39], [41]
Uncertainty management	Calibrated predicates improve uncertainty-aware planning in simulated manipulation [20]
Governance	Provenance, policy gates, reversibility, and bounded autonomy are explicit design targets [2]

Research Gaps and Unresolved Problems in Cognitive AI The Missing Empirical Link Between Component-Level Advances and General Cognitive Autonomy

The literature demonstrates progress in individual capabilities, but evidence that modules compose into general autonomy remains limited. Cognitive-architecture surveys identify hundreds of systems and many applications, yet no architecture provides human-level general intelligence [4], [8]. Robotics studies often validate a narrow task, while agent studies report benchmark success without establishing durable adaptation, transfer, or safe operation. The unresolved question is whether improvements in perception, planning, memory, and metacognition produce additive capability or new coordination failures.

Limited Standardization of Cross-Domain Benchmarks and Longitudinal Evaluation

Current evaluations vary in task definitions, environments, interfaces, success criteria, and reporting practices. A review of 15 agent benchmarks found that none incorporated safety or security into scoring, none used cost-efficiency metrics as a primary measure, and 13 relied exclusively on binary success measures [55]. These benchmarks therefore provide weak evidence about maintainability, resource use, recovery, and long-term learning. Evaluation should combine task success with trajectory quality, calibration, intervention frequency, computational cost, safety violations, and retention across distribution shifts [53].

Open Problems in Uncertainty Propagation, Grounded Knowledge, and Neural-Symbolic Coordination

Neural systems can infer useful representations from noisy data, whereas symbolic systems support explicit constraints and traceable inference, but interfaces between them remain difficult to scale [17], [18]. Perceptual uncertainty may be lost during symbol grounding, and knowledge bases may contain incomplete, contradictory, or outdated claims. Foundation models can reduce manual engineering for symbolic interfaces, yet end-to-end grounding faces combinatorial complexity [19]. Research therefore needs shared representations that preserve uncertainty, provenance, contradiction handling, and causal relevance across perception, reasoning, and action.

Safety, Regulation, Resource Requirements, and the Limits of Autonomous Self-Direction

Autonomy increases the consequences of incorrect goals, uncontrolled exploration, and silent model change. Governance must specify which actions require approval, which policies are hard constraints, how decisions are logged, and how systems revert after failure [2]. Resource demands also constrain deployment: richer memory, simulation, search, and verification increase latency and energy use, while compact architectures may sacrifice representational capacity [27]. Regulation cannot assume that benchmark competence establishes fitness for unsupervised operation. Self-direction should therefore remain bounded by authority, environment, reversibility, and independently tested safety conditions.

II. CONCLUSION

Synthesis of the Evidence for Hybrid Cognitive Architectures

The evidence supports architectures that combine learned perception, structured knowledge, memory, planning, and monitoring. Symbolic systems contribute explicit goals and constraints, while neural systems contribute robust extraction and adaptation. Robotic and uncertainty-aware studies show that integration can connect perception with action, but their restricted environments do not establish general autonomy.

Implications for the Design and Evaluation of Next-Generation Autonomous AI

Future agents should be evaluated as closed-loop systems operating across time. Their assessments must report reasoning quality, plan execution, adaptation, calibration, resource use, intervention, and failure recovery rather than prediction accuracy or binary task completion alone. Explainability should be treated as an inspectable causal record linking observations, knowledge, inferences, actions, and outcomes.

Priorities for Building Reliable Systems That Reason, Learn, Plan, and Act

Research priorities are shared state representations, uncertainty-preserving neural-symbolic interfaces, protected continual memory, metacognitive competence assessment, and governance integrated before deployment. Progress should be demonstrated through reproducible cross-domain tasks, longitudinal trials, ablations of architectural modules, and explicit safety and cost reporting. The target is therefore not unconstrained self-direction, but a cognitive agent whose reasoning and adaptation remain observable, interruptible, and accountable.

REFERENCES

- [1]. D. Vernon, G. Metta, and G. Sandini, "A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 2, pp. 151–180, Apr. 2007, doi: 10.1109/tevc.2006.890274.
- [2]. R. Haider, "Attention sovereignty in digital markets: Designing and testing consumer-owned preference agents as an alternative to surveillance-based advertising," Jul. 2026, doi: 10.13140/rg.2.2.27695.09125.
- [3]. F. Meneguzzi and G. Flucke, "Declarative planning in procedural agent architectures," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6508–6520, Jun. 2013, doi: 10.1016/j.eswa.2013.05.058.
- [4]. "European journal of prosthodontics and restorative dentistry," May 2026, doi: 10.1922/ejpr.v34i.
- [5]. M. Á. González-Santamarta, F. J. Rodríguez-Lera, C. Fernández, and V. M. Olivera, "A hybrid cognitive architecture to generate, control, plan, and monitor behaviors for interactive autonomous robots," *International Journal of Social Robotics*, vol. 17, no. 12, pp. 3135–3151, Dec. 2024, doi: 10.1007/s12369-024-01192-4.
- [6]. R. Haider *et al.*, "Forensic consumer psychology: An evidence-based framework for detecting deception and digital market manipulation," *Natural Resources for Human Health*, vol. 6, no. 16s, pp. 58–73, 2026.
- [7]. N. Muscettola, G. A. Dorais, C. Fry, R. Levinson, C. Plaunt, and D. Clancy, "IDEA: Planning at the Core of Autonomous Reactive Agents," *NASA Technical Reports Server (NASA)*, 2002.
- [8]. R. Haider, Md. R. Amin, Md. S. Arafat, I. Hossain, and R. Ahmad, "Smart Automation: Revolutionizing Business Operations, Advertising Costs and Customer Satisfaction Through Technology," *European Economic Letters*, vol. 16, no. 1, p. null, 2026.
- [9]. U. Nawaz, M. Anees-ur-Rahaman, and Z. Saeed, "A review of neuro-symbolic AI integrating reasoning and learning for advanced cognitive systems," *Intelligent Systems with Applications*, vol. 26, p. 200541, May 2025, doi: 10.1016/j.iswa.2025.200541.
- [10]. W. Wang, Y. Yang, and F. Wu, "Towards data-and knowledge-driven AI: A survey on neuro-symbolic computing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 2, pp. 878–899, Oct. 2024, doi: 10.1109/tpami.2024.3483273.
- [11]. D. Cunningham, M. Law, J. M. Lobo, and A. Russo, "The role of foundation models in neuro-symbolic learning and reasoning," *arXiv (Cornell University)*. Feb. 02, 2024, doi: 10.48550/arxiv.2402.01889.
- [12]. G. Allit, "EvoMind: A governed cognitive architecture for autonomous reasoning, planning, memory, and desktop agents," *Zenodo (CERN European Organization for Nuclear Research)*. Jun. 06, 2026, doi: 10.5281/zenodo.20580152.
- [13]. J. Wu and S. Yu, "A neuro-symbolic framework for reasoning under perceptual uncertainty: Bridging continuous perception and discrete symbolic planning," *arXiv (Cornell University)*. Nov. 18, 2025, doi: 10.48550/arxiv.2511.14533.
- [14]. RaiyanHaider, "Generative AI and quantum-inspired optimization: Redefining portfolio risk management and real-time capital allocation in volatile markets," *Journal of Informatics Education and Research*, Mar. 2026, doi: 10.52783/jier.v6i1.4540.
- [15]. M. Tenorth, D. Jain, and M. Beetz, "Knowledge processing for cognitive robots," *KI - Künstliche Intelligenz*, vol. 24, no. 3, pp. 233–240, Jun. 2010, doi: 10.1007/s13218-010-0044-0.
- [16]. O. Sychev, "Combining neural networks and symbolic inference in a hybrid cognitive architecture," *Procedia Computer Science*, vol. 190, pp. 728–734, 2021, doi: 10.1016/j.procs.2021.06.085.
- [17]. K. Thilagavathi, S. Suneel, R. Divya, G. Kavitha, G. D, and R. Naveenkumar, "Neuro-Symbolic agents for explainable autonomous reasoning in complex environments," in *2025 6th International Conference on Smart Electronics and Communication (ICOSEC)*, IEEE, Sep. 2025, pp. 1911–1917, doi: 10.1109/icosec67334.2025.11459656.
- [18]. R. Haider, T. Dwivedi, A. G. G. Girish, N. Verma, B. Kashyap, and V. S. Dubey, "Neuromarketing approaches to shaping healthy consumer choices: An integrative analysis of methods, efficacy, and ethical considerations," *CLOK (University of Central Lancashire)*, Mar. 2026, doi: 10.25258/ijddt.16.4s.62.
- [19]. Y. Zi, K. Roy, and A. Sheth, "Cognitive neurosymbolic artificial intelligence for complex decision-making: Integrating foundation models, cognitive architectures, and knowledge," *IEEE Intelligent Systems*, vol. 40, no. 3, pp. 16–22, May 2025, doi: 10.1109/mis.2025.3561487.
- [20]. Z. Wan *et al.*, "Towards cognitive AI systems: a survey and prospective on neuro-symbolic AI," *arXiv (Cornell University)*. Jan. 02, 2024, doi: 10.48550/arxiv.2401.01040.
- [21]. R. Haider, Md. F. I. Shaif, R. Ahmed, N. H. Nafi, M. R. Sumon, and M. Rahman, "Assessing the impact of influencer marketing on brand value and business revenue: An empirical and thematic analysis," *International Journal of Science and Research Archive*, vol. 16, no. 2, pp. 471–482, Aug. 2025, doi: 10.30574/ijrsra.2025.16.2.2355.
- [22]. M. LaValle, "Planning Algorithms," 2006.

- [23]. P. Langley, J. E. Laird, and S. Rogers, "Cognitive architectures: Research issues and challenges," *Cognitive Systems Research*, vol. 10, no. 2, pp. 141–160, Dec. 2008, doi: 10.1016/j.cogsys.2006.07.004.
- [24]. J. Lee *et al.*, "Hierarchical reinforcement learning with AI planning models," *arXiv (Cornell University)*. Mar. 01, 2022, doi: 10.48550/arxiv.2203.00669.
- [25]. R. Haider, Md. F. I. Shaif, R. Ahmed, N. H. Nafi, M. R. Sumon, and M. Rahman, "The influence of social media branding on consumer purchase behavior: A comprehensive empirical and thematic analysis," *International Journal of Science and Research Archive*, vol. 16, no. 2, pp. 460–470, Aug. 2025, doi: 10.30574/ijrsra.2025.16.2.2354.
- [26]. G. Konidaris, L. P. Kaelbling, and T. Lozano-Pérez, "From skills to symbols: Learning symbolic representations for abstract high-level planning," *Journal of Artificial Intelligence Research*, vol. 61, pp. 215–289, Jan. 2018, doi: 10.1613/jair.5575.
- [27]. M. Ghallab, D. Nau, and P. Traverso, "Automated planning and acting," *Cambridge University Press eBooks*. Jul. 31, 2016, doi: 10.1017/cbo9781139583923.
- [28]. R. Haider, J. T. Juba, and S. Chowdhury, "Exploring the link between suicidal ideation and digital environments: The hidden impact of marketing content," *International Journal of Science and Research Archive*, vol. 16, no. 2, pp. 607–614, Aug. 2025, doi: 10.30574/ijrsra.2025.16.2.2353.
- [29]. K. Miyazawa, T. Horii, T. Aoki, and T. Nagai, "Integrated cognitive architecture for robot learning of action and language," *Frontiers in Robotics and AI*, vol. 6, p. 131, Nov. 2019, doi: 10.3389/frobt.2019.00131.
- [30]. R. Haider, W. A. Megha, J. T. Juba, A. Alamgir, and L. Ahmad, "The conversational revolution in health promotion: Investigating chatbot impact on healthcare marketing, patient engagement, and service reach," *International Journal of Science and Research Archive*, vol. 15, no. 3, pp. 1585–1592, Jun. 2025, doi: 10.30574/ijrsra.2025.15.3.1937.
- [31]. R. P. N. Rao, D. C. Gklezakos, and V. Sathish, "Active predictive coding: A unifying neural model for active perception, compositional learning, and hierarchical planning," *Neural Computation*, vol. 36, no. 1, pp. 1–32, Dec. 2023, doi: 10.1162/neco_a_01627.
- [32]. R. Haider, F. A. I. Bari, Osru, N. Afia, and M. A. Khan Mugdho, "Leveraging internet of things data for real-time marketing: Opportunities, challenges, and strategic implications," *International Journal of Science and Research Archive*, vol. 15, no. 3, pp. 1657–1663, Jun. 2025, doi: 10.30574/ijrsra.2025.15.3.1936.
- [33]. T. Flesch, J. Balaguer, R. Dekker, H. Nili, and C. Summerfield, "Comparing continual task learning in minds and machines," *Proceedings of the National Academy of Sciences*, vol. 115, no. 44, pp. E10313–E10322, Oct. 2018, doi: 10.1073/pnas.1800755115.
- [34]. I. Kotseruba and J. K. Tsotsos, "40 years of cognitive architectures: core cognitive abilities and practical applications," *Artificial Intelligence Review*, vol. 53, no. 1, pp. 17–94, Jul. 2018, doi: 10.1007/s10462-018-9646-y.
- [35]. T. Flesch, A. Saxe, and C. Summerfield, "Continual task learning in natural and artificial agents," *Trends in Neurosciences*, vol. 46, no. 3, pp. 199–210, Jan. 2023, doi: 10.1016/j.tins.2022.12.006.
- [36]. S. Ho, M. Liu, L. Du, L. Gao, and Y. Xiang, "Prototype-Guided memory replay for continual learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 8, pp. 10973–10983, Mar. 2023, doi: 10.1109/tnnls.2023.3246049.
- [37]. A. Grandi, "A modular cognitive architecture for autonomous, self-motivated agents." Institute of Electrical and Electronics Engineers (IEEE), Feb. 03, 2026, doi: 10.36227/techrxiv.176462076.64363307/v2.
- [38]. A. I. Panov, "Simultaneous learning and planning in a hierarchical control system for a cognitive agent," *Automation and Remote Control*, vol. 83, no. 6, pp. 869–883, Jun. 2022, doi: 10.1134/s0005117922060054.
- [39]. S. Ko, "Logos: A cognitive architecture for unified reasoning and continual learning," *Zenodo (CERN European Organization for Nuclear Research)*. Apr. 09, 2026, doi: 10.5281/zenodo.19479043.
- [40]. R. Valiente and P. K. Pilly, "Metacognition for unknown situations and environments (MUSE)," *Neural Networks*, vol. 194, p. 108131, Sep. 2025, doi: 10.1016/j.neunet.2025.108131.
- [41]. M. Á. González-Santamarta, F. J. Rodríguez-Lera, and V. M. Olivera, "Cognitive architectures in autonomous robotics: A systematic review of behavior generation approaches and evaluation strategies," *IEEE Access*, vol. 13, pp. 191619–191644, 2025, doi: 10.1109/access.2025.3630344.
- [42]. M. A. Ali, F. Dornaika, and J. Charafeddine, "Agentic AI: a comprehensive survey of architectures, applications, and future directions," *Artificial Intelligence Review*, vol. 59, no. 1, Nov. 2025, doi: 10.1007/s10462-025-11422-4.
- [43]. P. Putta *et al.*, "Agent q: Advanced reasoning and learning for autonomous AI agents," *arXiv (Cornell University)*. Aug. 13, 2024, doi: 10.48550/arxiv.2408.07199.
- [44]. R. Haider, M. Bari, Osru, N. Afia, and T. Al Karim, "Illuminating the black box: Explainable AI for enhanced customer behavior prediction and trust," *International Journal of Science and Research Archive*, vol. 15, no. 3, pp. 247–268, Jun. 2025, doi: 10.30574/ijrsra.2025.15.3.1674.
- [45]. R. Haider, "The algorithmic memory of brands: Developing a longitudinal theory of how large language models learn, retain, distort and correct brand knowledge proposal field details," Jul. 2026, doi: 10.13140/rg.2.2.26017.36966.
- [46]. Cipher, "A four-tiered cognitive architecture for advanced AI reasoning," *Zenodo (CERN European Organization for Nuclear Research)*, Jun. 2025, doi: 10.5281/zenodo.15720539.
- [47]. R. Haider and J. Sabatina, "Harnessing the power of micro-influencers: A comprehensive analysis of their effectiveness in promoting climate adaptation solutions," *International Journal of Science and Research Archive*, vol. 15, no. 2, pp. 595–610, May 2025, doi: 10.30574/ijrsra.2025.15.2.1448.

- [48]. R. Haider, "Navigating the digital political landscape: How social media marketing shapes voter perceptions and political brand equity in the 21st century," *International Journal of Science and Research Archive*, vol. 15, no. 1, pp. 1736–1744, Apr. 2025, doi: 10.30574/ijrsra.2025.15.1.1217.
- [49]. A. Farooq, S. Raza, M. N. Karim, H. Iqbal, A. V. Vasilakos, and C. Emmanouilidis, "EVALUATING AND REGULATING AGENTIC AI: A STUDY OF BENCHMARKS, METRICS, AND REGULATION," *Information Fusion*, vol. 136, p. 104444, Oct. 2025, doi: 10.1016/j.inffus.2026.104444.
- [50]. R. Haider, M. Bari, Md. F. I. Shaif, and M. Rahman, "Engineering hyper-personalization: Software challenges and brand performance in AI-driven digital marketing management: An empirical study," *International Journal of Science and Research Archive*, vol. 15, no. 2, pp. 1122–1141, May 2025, doi: 10.30574/ijrsra.2025.15.2.1525.
- [51]. T. Kehkashan *et al.*, "From benchmarks to deployment: a comprehensive review of agentic AI evaluation," *Artificial Intelligence Review*, vol. 59, no. 8, Apr. 2026, doi: 10.1007/s10462-026-11571-0.
- [52]. R. Haider, M. Bari, Md. F. I. Shaif, M. Rahman, Md. N. H. Ohi, and K. M. M. Rahman, "Quantifying the impact: Leveraging AI-Powered sentiment analysis for strategic digital marketing and enhanced brand reputation management," *International Journal of Science and Research Archive*, vol. 15, no. 2, pp. 1103–1121, May 2025, doi: 10.30574/ijrsra.2025.15.2.1524.
- [53]. P. Langley and D. Choi, "A unified cognitive architecture for physical agents," pp. 1469–1474, 2006.
- [54]. J. S. Albus and A. J. Barbera, "RCS: A cognitive architecture for intelligent multi-agent systems," *Annual Reviews in Control*, vol. 29, no. 1, pp. 87–99, 2005, doi: 10.1016/j.arcontrol.2004.12.003.
- [55]. H. Chong, A. Tan, and G.-W. Ng, "Integrated cognitive architectures: a survey," *Artificial Intelligence Review*, vol. 28, no. 2, pp. 103–130, Aug. 2007, doi: 10.1007/s10462-009-9094-9.
- [56]. R. Haider, "From search visibility to answer share: Developing and testing a causal theory of brand discoverability and consumer conversion in generative engine optimization," Jul. 2026, doi: 10.13140/rg.2.2.24339.64807.