<Cognitive Architectures and General Intelligent Systems>
- by Pat Langley -

Presented by Yoo, Jung Sik
Outline

- Why General?
- Three Architectural Paradigms
- The Architecture of the ICARUS
  - Memories and Representation
  - Performance and Learning Process
  - Architectures as Programming Languages
- Conclusion
1. Why General?
Why General?

- The original goal of artificial intelligence was the design and construction of computational artifacts that combined many cognitive abilities in an integrated system.

- Modern artificial intelligence has largely abandoned this objective.

- Within each of its subfields there was some progress but the most researches seldom show concern about how the separate modules might interact with each other.
"You can't play twenty questions with nature and win."

- A famous argument by Allen Newell (1973)
- The statement was originally criticizing the strategy of experimental cognitive psychologists, who studied isolated components of human cognition without considering their interaction.
- Over the past decade, the statement has become an equally valid criticism of the fragmented nature of AI research.

Allen Newell (1927-1992)
“We should evaluate the systems in terms of generality and flexibility, rather than success on a single domain.”
2. Three Architectural Paradigms
Three architectural paradigms

- the *multi-agent systems* framework (Sycara 1998)
  - has much in common with traditional approaches to software engineering.
  - Distinct modules for different facets of an intelligent system, which then communicate directly with each other.
  - The inputs/outputs and the protocols for communications are specified by the architecture.
  - A disadvantage of the multi-agent systems framework is the need for modules to communicate directly with one another.
Three architectural paradigms

- **blackboard system** (Engelmore and Morgan 1989)
  - replaces direct communication among modules with an indirect scheme that relies on matching patterns against elements in the short-term memory.
  - resembles human cognition in real sense.
  - Maybe the better representation of human-like AI than the multi-agent system.
More than the previous two

- A. Newell believed that agent architectures should incorporate strong theoretical assumptions about the nature of the mind.

- **4 commitments in AI research**
  - 1. The short-term and long-term memories that store the agent’s beliefs, goals, and knowledge
  - 2. The representation and organization of structures that are embedded in these memories
  - 3. The functional processes that operate on these structures, including both performance and learning mechanisms
  - 4. The functional processes that operate on these structures, including both performance and learning mechanisms
3. The Architecture of ICARUS

3.1. Memories and Representations
The **ICARUS Architecture**

- The five high-level principles about the nature of general intelligent systems.
  - Cognition is grounded in perception and action
  - Concepts and skills are distinct cognitive structures
  - Long-term memory is organized in a hierarchical fashion
  - Skill and concept hierarchies are acquired in a cumulative manner
  - Long-term and short-term structures have a strong correspondence
Memories and Representations

**Memories and Representations**

- *ICARUS* includes six distinct memories, two of each variety.
- Two separate long-term memories, one for conceptual knowledge and another for skills or procedures.
- The framework also has two analogous short-term memories, one for the agent’s beliefs about the environment and another for its goals and associated intentions.
- *ICARUS* has a perceptual buffer that holds immediate perceptions of the environment and a motor buffer that contains skills and actions it intends for immediate execution.
Memories and Representations

- Most frameworks rely on formalisms similar to the predicate calculus that support expression of relational content.
- Both type of long-term memories are organized in hierarchical terms, with more complex skills and concepts being defined in terms of simpler components.
- *ICARUS*’s short-term belief memory contains instances of defined concepts, which encode specific beliefs about the environment that the agent can infer from its perceptions.
- This contains a prioritized set of goal stacks, each of which contains an ordered list of goals, with an entry serving as the subgoal for the one below it on the list.
3. The Architecture of ICARUS

3.2. Performance and Learning Processes
Performance and Learning Processes

Performance and Learning Processes

- Performances typically include methods for memory retrieval, pattern matching, skill selection, inference, and problem solving.

- Learning processes are responsible for altering the contents of long-term memory, either by generating new knowledge structures or by refining and modulating existing structures.
Performance and Learning Processes

- On each cycle conceptual inference of, the system matches concept definitions in long-term memory against perceptions and beliefs. When a concept matches, the module adds an instance of that concept to short-term belief memory, making it available to support other inferences.

- When a concept matches, the module adds an instance of that concept to short-term belief memory, making it available to support other inferences.

- The system operates in a bottom-up manner, starting with primitive concepts.
In contrast with conceptual inference, the skill execution module proceeds in a top-down manner.

The process starts from the current goal and finds applicable paths through the hierarchy that terminate in primitive skill clauses with executable actions.

When ICARUS’s execution module can’t find a path through the skill hierarchy relevant to its current goal, it invokes a module for means-ends problem solving.

ICARUS incorporates a learning module that creates a new skill whenever problem solving and execution achieve a goal.
3. The Architecture of ICARUS

3.3. Architectures as Programming Language
Architectures as Programming Language

- A cognitive architecture typically comes with an associated programming language for use in building knowledge-based systems.
- The programming language associated with ICARUS comes with the syntax for hierarchical concepts and skills, the ability to load and parse such programs, and commands for specifying the initial contents of short-term memories and interfaces with the environment.
- The language also includes an interpreter that handles inference, execution, planning, and learning over these structures, along with a trace package that displays system behavior on each cycle.
4. Conclusion

- “Architectural research is all about mutual constraints, and its aim should be a unified theory of intelligent behavior, not merely an integrated one.” - A. Newell (1990) -

- According to Newell it should follow the four commitment which are system's memories, the representation of those memories’ contents, the functional processes that operate on those contents and the programming language.

- ICARUS is such an architecture.

- The AI studies would progress with increased research on topics that have received little attention within traditional cognitive architectures.
Thank you.