An Adaptive Architecture for Radically Autonomous Systems

Pat Langley
Institute for the Study of Learning and Expertise

Award Period 7/1/15–6/30/18

ONR Cognitive Science and Human-Robot Interaction

Program Review

August 4-7, 2015

Research Objectives

We aim to develop an architecture for intelligent agents that supports radical autonomy by:

- Annotating symbolic goals with numeric priorities
- Incorporating both achievement and maintenance goals
- Using partial-satisfaction planning to handle many goals
- Conditioning strategic control on problem characteristics

The research project's significance lies in its potential to:

- Enable more fully autonomous robotic and software agents
- Improve our understanding of autonomy in humans

The effort will combine ideas from different traditions in a unified account of goal-directed behavior.

The Nature of Autonomy

Truly autonomous agents could aid the US military, and other facets of our society, in many ways.

We will say that a computational system is *autonomous* when it:

- Operates in some environment *over time*
- Responds *adaptively* to its situation
- Selects *which actions to carry out*
- Decides *how to allocate resources*
- Determines *which goals to pursue*

Such an agent may interact with others and be influenced by them, but it makes its own decisions.

Humans exhibit substantial autonomy, and we would like to reproduce their ability in machines.

An Autonomy Scenario

Consider a robotic agent on an extended mission with many goals – of different priorities – that involve:

- Carrying out specified tasks
- Achieving specified states
- Maintaining certain conditions

Environmental uncertainty, action reliability, and task urgency can vary, requiring different strategies for planning and execution.

The robot may need to team with others and thus make decisions about communicating and coordinating with them.

We desire a computational theory that supports all these abilities.

Previous Work on Autonomy

Prior AI research has explored some limited forms of autonomy, including:

- Reactive control systems (Nilsson, 1994; Parker, 1995)
- AI planning systems (Ghallab, Nau, & Traverso, 2004)
- Cognitive architectures (Langley, Laird, & Rogers, 2009)

Work on *goal reasoning* (Aha et al., 2013) has gone further, but developers still write rules for generating top-level goals.

We propose to develop an agent architecture for more radical autonomous systems.

Architectural Assumptions

We will build on the HPS and HPE modules, developed in the previous project, which assume that:

- Plans recursively decompose problems into subproblems
- Search is organized as an OR tree, each node elaborating its parent
- Planning and execution each cycle through five stages
- Domain-independent knowledge determines the decisions at each stage, including when to generate / execute plans
- Domain-specific knowledge states ways to decompose problems, playing the same role as domain operators

This framework supports a wide range of goal-oriented behaviors, but still has some important limitations.

We will extend it in four ways to provide greater agent autonomy.

Aim 1: Goal Priorities and Motives

Most previous research on problem solving has adopted one of two schemes:

- Symbolic goals that describe desired states
- Numeric functions that describe distance to target

We will extend HPS to associate numeric priorities with symbolic goals, unifying the two paradigms.

These numeric scores map onto earlier notions of *motivation* in psychology, but they link to cognitive elements.

We will also include long-term structures – *motives* – that specify conditions for activating top-level goals.

Aim 1: Goal Priorities and Motives

We must also provide HPS with processes that operate over this extended representation by:

- Using goal priorities to select a node (partial plan) to elaborate next
- Using goal priorities to choose operators to add to partial plans
- Calculating priorities of subgoals in new subproblems produced by backward chaining
- Examining goal priorities to decide if a problem / subproblem has been 'solved' (e.g., if weighted sum exceeds a threshold).

We can add these abilities as strategic knowledge; plan execution should require even fewer changes.

HPS must also recalculate goal priorities as the situation changes, using numeric functions attached to motivational rules.

Aim 2: Achievement / Maintenance Goals

Research on plan generation has emphasized *achievement* goals; work on plan execution has focused on *maintenance* goals.

HPS supports the former, but we will extend it to handle both by:

- Associating start and end times with beliefs, goals, and intentions
- Denoting when beliefs become true (false) with start (end) times
- Giving achievement goals unspecified start times (in the future) and unspecified end times
- Giving maintenance goals constrained start times and end times

We can view an agent's goals as a *temporal constraint network* (Barrett et al., 2004; Ingham et al., 2005).

From this perspective, achievement and maintenance goals differ only in their temporal constraints.

Aim 2: Achievement / Maintenance Goals

We must also extend the current architecture's mechanisms to utilize these elaborated goal structures:

- Redefine goal satisfaction over temporal intervals
  - For subgoals on operator conditions before execution
  - For subgoals of defined predicates during inference
- Criteria for matching operator conditions, effects, and goals
- Local plan repair rather than recreation from scratch

Goal priorities should focus the agent's attention when multiple issues arise.

Aim 3: Handling Many Goals

Studies of human problem solving have emphasized constrained tasks that involve only a few goals.

Classic work on reactive control (Ingrand et al., 1992) dealt with multiple maintenance goals, but handled them one at a time.

We will extend HPS to support partial satisfaction planning by:

- Altering the problem solver's termination criteria
- Deciding when unexpected events deserve attention
- Improving revision of plans when required for execution

Again, we will not need to alter the basic architecture; changes to strategic knowledge should suffice.

Aim 4: Adaptive Planning / Execution

The planning/execution literature has reported many techniques:

- Forward vs. backward search
- Best-first vs. beam vs. greedy search
- Closed-loop vs. open-loop control

Strategic knowledge can encode these alternatives in HPS, but we posit their effectiveness depends on task features.

- E.g., whether forward or backward search is better depends on relative branching factors.

We will identify such features, define relevant predicates, and include them in conditions on strategic control rules.

Aim 4: Adaptive Planning / Execution

We must also modify the architecture's mechanisms to use the enriched control rules; this will involve:

- Finding matched rules and selecting one for application
- Calculating meta-level problem features used in matching:
  - Static (e.g., whether an operator is reversible)
  - Changing gradually (e.g., estimates of operator reliability)
  - Highly dynamic (e.g., relative branching factor)

The architecture must calculate such features from information available in the current search node.

This idea has much in common with work on meta-cognition and meta-level control (Cox, 2005).

Relation to Psychological Findings

The planned architecture makes contact with key phenomena and concepts from psychology, in that it:

- Unifies notions of structural goals and motivations
- Mimics human capacity for achievement and maintenance goals
- Supports satisficing on tasks with many goals / criteria
- Accounts for variation / adaptability of human problem solving

Other frameworks, like Soar and ACT-R, address these issues but make few theoretical, architecture-level commitments about them.

Plans for Evaluation

We will evaluate the expanded architecture along a number of dimensions:

- Use simulated environments to demonstrate new abilities
- Run on both single-agent and coordinated multi-agent scenarios
- Use controlled experiments to show benefits of extensions
- Document consistency with major psychological findings

Together, these analyses should show that the new architecture supports human-like autonomy with practical relevance.

Summary Remarks

We are extending our framework for flexible problem solving and execution to:

- Annotate symbolic, relational goals with numeric priorities
- Reason about both achievement and maintenance goals
- Focus attention on a subset of many simultaneous goals
- Adapt strategic control depending on features of problems

Together, these should produce an integrated architecture that supports creation of radically autonomous agents.

Transition Plan

Our research on architectures for intelligent agents has clear uses in mobile robots and shipboard autonomy.

In the longer term, we hope to transition our results to applied settings like:

- Cognitive robots that interact with Navy personnel dealing with shipboard problems (e.g., fighting fires)
- Intelligent ships that offload many operational details currently handled by human experts.

We hope to take advantage of existing relationships with NRL researchers to increase the chances of successful transitions.

Project Budget

The research project's budget, by federal fiscal year, is:

- FY2016: $ 40K [?]
- FY2017: $180K [?]
- FY2018: $183K [?]
- FY2019: $140K [?]

No DURIP are being awarded in relation to this project.

End of Presentation