Second Annual Conference on Advances in Cognitive Systems Poster Collection (2013) 29-45

Goal-Driven Autonomy and Question-Based Problem Recognition

Michael T. Cox MCOX@CS.UMD.EDU
Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 USA

Abstract

Autonomy involves not only the capacity to achieve the goals one is given but also to recognize problems and to generate new goals of one's own that are worth achieving. The creation of goals starts with the recognition of a novel problem, and this recognition begins with the detection of anomalies represented as the difference between expectations and observations (or inferences). The expectation failure triggers the posing of questions; questions lead to explanations; and explanations form the basis for goals. I illustrate these principles with examples from a computational architecture called MIDCA.

1. Introduction

> Have patience with everything that remains unsolved in your heart. Try to love the questions themselves, like locked rooms and like books written in a foreign language. Do not now look for the answers. They cannot now be given to you because you could not live them. It is a question of experiencing everything. At present you need to live the question. Perhaps you will gradually, without even noticing it, find yourself experiencing the answer, some distant day. ― *Letter 4* (Rilke, 1986/1903)

Certainly one of the key elements of intelligence that separates us from the rest of the primate order is that we routinely question our world and ourselves. Yet some questions can take us in false directions and waste valuable time and effort. Other questions are rhetorical and serve quite different functions (e.g., speech acts). However I claim here that *questions frame an inquiry that lies at the heart of what it means to be intelligent*. Although the performance of the individual may benefit, it is not the efficiency of action that primarily motivates questioning and answering. Instead it is the agent in search of truth that operates behind the curtain. In seeking to understand the world and ourselves, we discover problems in our understanding of and in our world. By addressing these problems, we can better comprehend and independently manage the worlds within which we exist. Represented appropriately, these problems give rise to goals that enable change in our environment as well as change within ourselves.

Goals are key computational structures that drive problem solving, comprehension, and learning. In humans they constitute a special category of knowledge structure that represents desired and attainable states of affairs and that holds some measure of value for the individual (Kruglanski, 1996). For Kruglanski goals possess properties attributable to all knowledge, to goals as a class, and to the specific goal relation identified. Hierarchical systems of goal structures motivate and constrain reasoning and effective behavior (Kruglanski, Köpetz, Bélanger, Chun, Orehek, & Fishbach, 2013; Kruglanski, Shah, Fishbach, Friedman, Young, &

© 2013 Cognitive Systems Foundation. All rights reserved.

Chun 2002). These goal systems enable efficient self-regulation and self-control. Goal orientation has also been shown to be an important determinant for successful learning (Dweck, 1986). Indeed the claim has been made that virtually all human behavior involves the pursuit of goals, that goals are at the center of human learning, and thus that goals constitute an effective means by which to organize educational curricula (Schank, 1994).

In artificial cognitive systems, goals serve functions similar to those they do in people. In both cases goals provide focus, direction, and coordination for the allocation of computational resources during problem-solving and inference making. They also form a basis for the organization of and retrieval from memory (Schank & Abelson, 1977; Schank, 1982). Computationally goals reduce the amount of processing involved in decision-making; psychologically they afford attention and provide purpose. Further they also provide the means for explaining behavior (Malle, 2004; Ram, 1990; Schank, 1982; 1986). That is, agents do particular actions because they intend to achieve the states that result from such actions. Finally, many researchers have shown a positive relation between goal-focused behavior and deliberate, planful learning (Cox, 2007; Cox & Ram, 1999; Ram & Leake, 1995).

However most computational theories of cognition assume the existence of goal structures and concentrate on their application and use in problem-solving, planning and execution, learning and other activities. Except under two conditions, these theories generally do not account for how goals originate. One assumption is that goals are simply input by a human. A second possibility is that goals arise due to subgoaling. When action preconditions are unsatisfied in the environment, a sub-goal is spawned to achieve blocked preconditions. Here I will examine in detail an alternative cognitive process that creates novel goals for an autonomous agent.

Autonomy involves not only the capacity to achieve given goals, but it also concerns the ability to recognize new problems and to propose goals that are worth achieving. An independent cognitive system is autonomous to the extent that it understands the world and its relation to it and can act accordingly. Both opportunities and threats in the world require an agent to anticipate events within the context of its interests. Given an understanding of these opportunities and threats, an autonomous agent manages the goals and plans it has by adapting either its plans or goals and by abandoning old goals or by generating new ones.

This paper will examine these issues in the context of computational theories and will illustrate parts of our theory with various examples. The following section discusses what it means for a cognitive system to be autonomous, and it will challenge the current models. Here autonomy means being able to generate new goals rather than just following the goals of others. The subsequent section expands upon this model and proposes that new goals come from the recognition of problems in the environment and within one's knowledge and experience. The key is to identify when observations violate expectations, to ask why such an anomaly occurred, to answer by explaining the causes of the anomaly, and to generate a goal to remove the primary cause. This process either points to a deficiency of knowledge or a deficiency in the world. Next we examine an architecture called MIDCA that implements many of these ideas and processes and that provides a direction for our research. I close with summary and concluding remarks.

2. Autonomy

A common model of autonomy assumes an intelligent agent that can perform tasks given to it by a user and/or can automatically improve its performance in some environment over time (Maes, 1994; Wooldridge, 2002). The consensus asserts that agents exist within some environment

(either real or virtual) and can both sense and act within that environment (Franklin & Graesser, 1997; Weiss, 1999). Weiss (1999) distinguishes autonomous agents from mere programs, since agents decide whether or not to perform a request; whereas called functional programs have no say in the matter. Many researchers attempt to restrict autonomous agents to those that have their own agenda or otherwise do not need human intervention to perform some task (e.g., Franklin & Graesser, 1997). However this transfers the uncertainty from the question of "what is autonomy?" to "what is having an agenda?" In general, an autonomous agent is one that makes many of its own decisions but that has an explicit mechanism of control. Control is provided by human assigned tasks or goals or by programming and design specifications. Autonomy is provided procedurally by automated planning and learning mechanisms.

2.1 Goal-Following Autonomy

Consistent with these characterizations, I define *goal-following autonomy* as either hardware or software agents that can accept a goal from a user (or another agent) and can automatically achieve the goal by performing a sequence of actions in its environment.[1] The goal is simply some state configuration of the environment in terms of what needs to be satisfied. Functionally the agent implements the following characteristics.

- A means to accept a goal request for some environment;
- A decision whether to pursue the goal given its current situation and any prior requests;
- A capacity to plan for goal achievement;
- A capability to interact with the environment by executing a plan and perceiving results.

If the world does not cooperate, a flexible agent may even change its plan at execution time to accomplish the goal as specified. But once the goal is achieved, does the agent wait to be told what to do next? Does it halt? In either case, this does not appear to constitute full autonomy in the more substantial sense.

Practical problems also exist with the goal-following model. Most autonomous unmanned vehicles (AUVs) control their behaviors through preprogrammed mission plans that specify a set of waypoints, vehicle parameters, and tasks to perform at particular waypoints (Hagen, Midtgaard, & Hasvold, 2007). Complex tasks that cannot be fully specified in advance rely upon sporadic human intervention and communication. A large portion of the existing research effort into physical platforms revolves around motion planning and obstacle avoidance (e.g., see Berry, Howitt, Gu, & Postlethwaite, 2012; Minguez, Lamiraux, & Laumond, 2008). However an unrealistic burden on the user exists if a human must continuously monitor the behavior of an AUV. Furthermore in situations of low bandwidth or stealth, continuous human to machine communications cannot be assumed. Yet in many complex, dynamic environments, unusual situations arise on a regular basis as the world changes in unexpected ways. This quandary has been called *the brittleness problem* (Duda, & Shortliffe, 1983; Lenat & Guha, 1989). That is, agents and virtually all cognitive systems in complex environments are brittle except in narrow situations that have been foreseen and verified previously by the system designers. But problems will occur, and a truly autonomous system should be robust in the face of surprise.

[1] The goal-following model of autonomy also includes those agents that can accept tasks to perform rather than states to achieve. In either case, the agent takes some high-level description of the desired behavior and automatically computes how to instantiate the directive in a particular environment.

One widespread solution to the brittleness problem is machine learning technology (Maes, 1994; Holland, 1986; Stone & Veloso, 2000). Instead of explicitly programming an agent to select the optimal choice among large numbers of candidates across all possible situations, machine learning attempts to create generalizations for those conditions that apply to the largest set of possible contingencies. For example a classifier maps from a state of the world to some choice or decision outcome. In the case of AUVs, this choice could be in terms of a particular action to perform given the current state. Learning from experience then would develop new responses for unexpected situations. Much success has of course been reported in the machine learning and agent literature and also in the cognitive systems community (e.g., see Li, Stracuzzi, & Langley, 2012 and Laird, Derbinsky, & Tinkerhess, 2012 for the latter).

I do not dispute that autonomy is about choosing good actions given a goal and about flexibly learning to improve these choices. But the goal-following model is not complete, and it misses an important distinction when the environment is extremely uncooperative and complex. Interestingly DARPA (2012, p. 8) has pointed to this missing component of autonomy in its capabilities description for its latest X-ship project called the ASW Continuous Trail Unmanned Vessel or ACTUV. Among the many autonomous capabilities, they identified that ACTUV needed to be "capable of autonomous arbitration between competing mission and operating objectives based on strategic context, mission phase, internal state, and external conditions." That is, autonomy implies the need to balance its own condition in relation to what is happening in the world with the strategic context of what it is trying to accomplish.

2.2 Goal-Driven Autonomy (GDA)

Broadly construed, the topic of *goal reasoning* concerns cognitive systems that can self-manage their goals (Vattam, Klenk, Molineaux, & Aha, in press). The topic is about high-level management of goals, plans, knowledge, and activities, not simply goal pursuit. In particular we focus on managing goals in the midst of failure. Failure is important for any cognitive system if it is to improve its performance in complex environments. First no programming no matter how extensive will guarantee success in non-trivial domains. Second failure points to those aspects of the system that are no longer relevant or contain some gap that needs filling in the new context. In particular cognitive systems (especially humans) generate expectations about what will or should occur in the world. Expectation failures drive much of cognition and a number of researchers have recognized this relationship (Anderson & Perlis, 2005; Birnbaum, Collins, Freed, & Krulwich, 1990; Cox & Ram, 1999; Perlis, 2011; Schank, 82; Schank & Owens, 1987).

The alternative model of autonomy I advocate here consists of self-motivated processes of generating and managing an agent's own goals in addition to goal pursuit. This model - called *goal-driven autonomy (GDA)* (Cox, 2007; Klenk, Molineaux, & Aha, 2013; Munoz-Avila, Jaidee, Aha, & Carter, 2010) - casts agents as independent actors that can recognize problems on their own and act accordingly. Furthermore in this goal-reasoning model, goals are dynamic and malleable and as such arise in three cases: (1) goals can be subject to transformation and abandonment (Cox & Veloso, 1998; Talamadupula, et al., 2010); (2) they can arise from subgoaling on unsatisfied preconditions (e.g., see Veloso, 1994) or in response to impasses (Laird, 2012; Laird, Rosenbloom, & Newell, 1986) during problem-solving and planning; and (3) they can be generated from scratch during interpretation (Cox, 2007; see also Norman, 1995).

For our purposes here, the most important of the above three cases is the third one. The idea is that given a problem in the world, an autonomous cognitive system must distinguish between

perturbations that require a change in plans for the old goal and those that require a new goal altogether. What is missing in the planning and agent communities is a recognition that autonomy is not just planning, acting and perceiving. It also must incorporate a first-class reasoning mechanism that interprets and comprehends the world as plans are executed (Cox, 2011). It is this comprehension process that not only perceives actions and events in the world, but can recognize threats to current plans, goals, and intentions. I claim that a balanced integration between planning and comprehension leads to agents that are more sensitive to surprise in the environment and more flexible in their responses.

In my approach, flexibility is realized through a technique I call *goal insertion*, where an agent inserts a goal into its planning process. In general goals are produced through *goal formulation* (c.f., Hanheide et al., 2010; Wilson, Molineaux, & Aha, 2013; Weber, Mateas, & Jhala, 2010), the process that includes the creation and deployment of autonomous goals where the agent takes initiative to establish new concerns and to pursue new opportunities. First an agent detects discrepancies between its observations and its expectations. The agent then explains what causes such discrepancies and subsequently generates a new goal to remove or mitigate the cause (Cox, 2007).

Consider a baby that is crying in public. Given that it normally is quite calm, the crying observation violates the mother's expectation and represents an anomaly. The baby's behavior could be explained in a couple of simple ways. If the baby is hungry then it cries, and if the baby's diapers are dirty then it cries. The mother may check the diapers to eliminate that explanation and conclude that it is hungry. The resulting goal for the mother is to eliminate the hunger. A reasonable plan would be to get a bottle from her bag and feed the baby. Table 1 shows the distinct stages of the GDA process in an overly simplified manner.

Table 1. Crying baby example

Steps	**Representations**
Anomaly detection	Expect: calm (baby) Observe: cry (baby)
Anomaly explanation	hungry (baby) → cry (baby)
Goal formulation	¬ hungry(baby)
Plan	get (bottle) feed (baby, bottle)

In this example, the explanation is a basic rule, but in general it may be an arbitrary lattice or explanatory graph structure. Here the goal is just the negation of the rule antecedent (see Cox, 2007 for an accompanying algorithm), but for realistic situations, goal formulation must choose some salient prior state or set of states as the cause that requires attention for the problem to be solved. The next section will take a closer look at this issue.

3. Question-Based Problem Recognition

Years ago Getzels (Getzels & Csikszentmihalyi, 1975) described the early stages of problem solving as including a cognitive process they called *problem finding* (see also Runco & Chand, 1994). Problem finding involves the identification of the problem and precedes problem-

representation. Getzels classified problem finding into three distinct classes: problem presentation involves tasks given to the subject; problem discovery is the detection or recognition of the problem; and problem creation is a creative act that designs new problems.[2] This paper focuses on the second class and uses the more common term *problem recognition* (Pretz, Naples, & Sternberg, 2003).[3]

In a brief *Cognitive Science* article, Getzels (1979) notes that, despite the journal's broad coverage of cognitive processes and the eminent role that problems play in such processes, little research examines how problems are found or formulated. Surprisingly this negative condition is still true today. Despite the acknowledgement that goal formulation constitutes an initial stage of computation for agents (e.g., Russell & Norvig, 1995, p. 56), authors and AI researchers still fail to investigate its operation. Instead the technical focus is on high quality problem-representation (by researchers) and optimal problem-solving algorithms (by machines). Computational problems and goal states are typically assumed as given (presumably by an intelligent human). But our claim here is that the recognition of the problem and the generation of a goal are the hard problems rather than the generation of solutions.

The GDA approach starts with an expectation failure as an initial condition for anomaly detection. However not all anomalies are problems and not all problems are relevant to the agent. Furthermore the representation for problems is often overly simplified in the literature. For example the automated planning community represents problems as an initial state, a goal state, and a set of operators (i.e., action models) (see Ghallab, Nau, & Traverso, 2004). But the real issue with how problem instances are represented is that in general they are arbitrary. Consider the blocksworld domain. The initial states in this domain are random configurations of blocks, but so too are the goals states. For example in the first panel of Figure 1, the initial state is the arrangement of three blocks on the table and the goal state is to have block A on top of block B. The planner is given no reason for why this should be the case, unlike the second panel in the figure. In this situation the planner wishes to have the triangle D on the block A to keep water out. Here D represents the roof of the house composed of A, B, and C. Water getting into a person's living space is a problem; stacking random blocks is not.

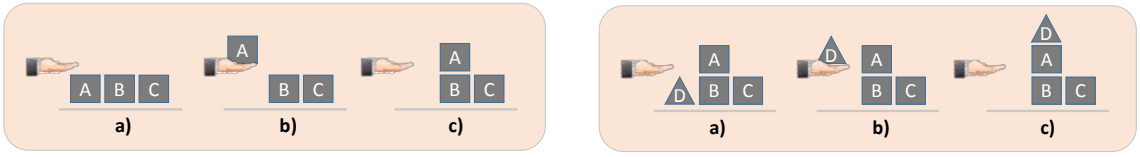

Figure 1. Blocksworld state sequences that distinguish justified problems from arbitrary problems

[2] Although none of these researchers take a computational approach, Hawkins, Best, & Coney (1989) actually come closest to our conception of the processes of problem recognition, but they do so from the perspective of a business analysis of consumer behavior. Their process starts with a discrepancy between what the consumer expects and what they perceive. Such problem recognition then leads to a solution in terms of a product to purchase. As an aside, it is interesting to note that it is in the interest of businesses and marketing firms to manufacture a consumer need by giving the consumer specific expectations. In this sense product marketing may be viewed as problem creation in Getzels classification.

[3] See also Klein, Pliske, Crandall, & Woods (2005) for *problem detection*. Their perspective is similar to ours (although still not computational), but they insist that the detection is not solely about expectation violations.

In our earlier work on symbolic anomaly detection (Cox, Oates, Paisner, & Perlis, 2012), we simulated an anomaly by removing an operator from the set of action models for the logistics domain. The logistics problems were to transport packages from one location to another using planes and trucks. When the unload-airplane action was removed and the original problem set was presented to the planner, only a subset of problems could be solved. The new set of solutions then provided an anomalous series of observations relative to the normal set of solutions. Now consider how a GDA-based cognitive system might actually explain the planes not being unloaded at a particular airport.

Figure 2 shows a possible explanation. The airplanes are not unloaded, because no workers are at the airport. This is the case because the stevedores are on strike. The strike is caused by bad working conditions and low pay. For the management of the logistics company, this represents a serious and relevant problem. If it continues, profits will plummet and investor confidence will wane. Both are thus threatened. The question remains as to the management's goal however.

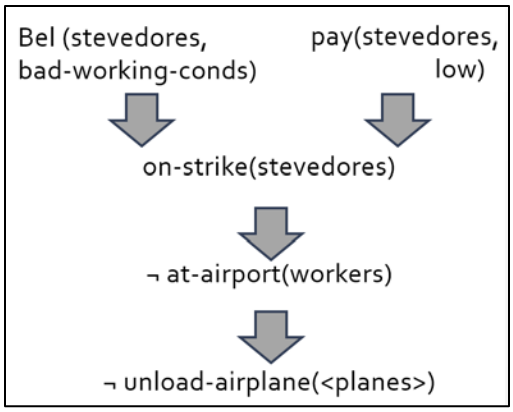

Figure 2. Explanation for why airplanes are not unloaded

Negating the pay predicate leads to higher wages; whereas negating the belief in poor conditions possibly leads to misinformation. An alternative is to negate ¬at-airport(workers) which is to imply workers should be present. This could be achieved by bringing in scabs. Finally the management might address the actual working conditions. In this last case, the goal would be ¬bad-working-conds. The choice of which cause of the problem to attack is unclear. A cost benefit analysis might be used to differentiate between choices, but one must be sure not to require a cost or benefit of the resulting plan or solution, only the goal. Otherwise one would need to perform planning before goal formulation – a clear case of the cart before the horse.

Instead an intelligent agent should ask the right question in the first place. It observes no planes unloading when it expects normal activity at the airport and experiences an expectation failure and hence an anomaly. If it asks why the planes were not unloaded it would get an explanation of how this was not the case. But if it asks why the workers chose not to perform their duty, it would not get an explanation that focused on their wages. They have unloaded planes in the past at these same wages. The answer would focus on the working conditions. The agent would then recursively question whether it was the perception of the conditions that changed (i.e., the belief) or the conditions themselves that changed. This secondary question would get to the core cause of the problem. Given an answer to the question and an appropriate explanation, the goal would be simple. Removing the cause of the grievance would be the goal that the autonomous agent should prefer. Planning and hence action would follow directly and effectively.

In the next section I will examine a cognitive architecture that integrates problem-solving as planning and comprehension as goal-driven autonomy. The architecture includes a cognitive cycle for action and perception and an analogous metacognitive cycle for meta-level control and introspective monitoring. I will concentrate on the former cycle to illustrate the concepts presented above.

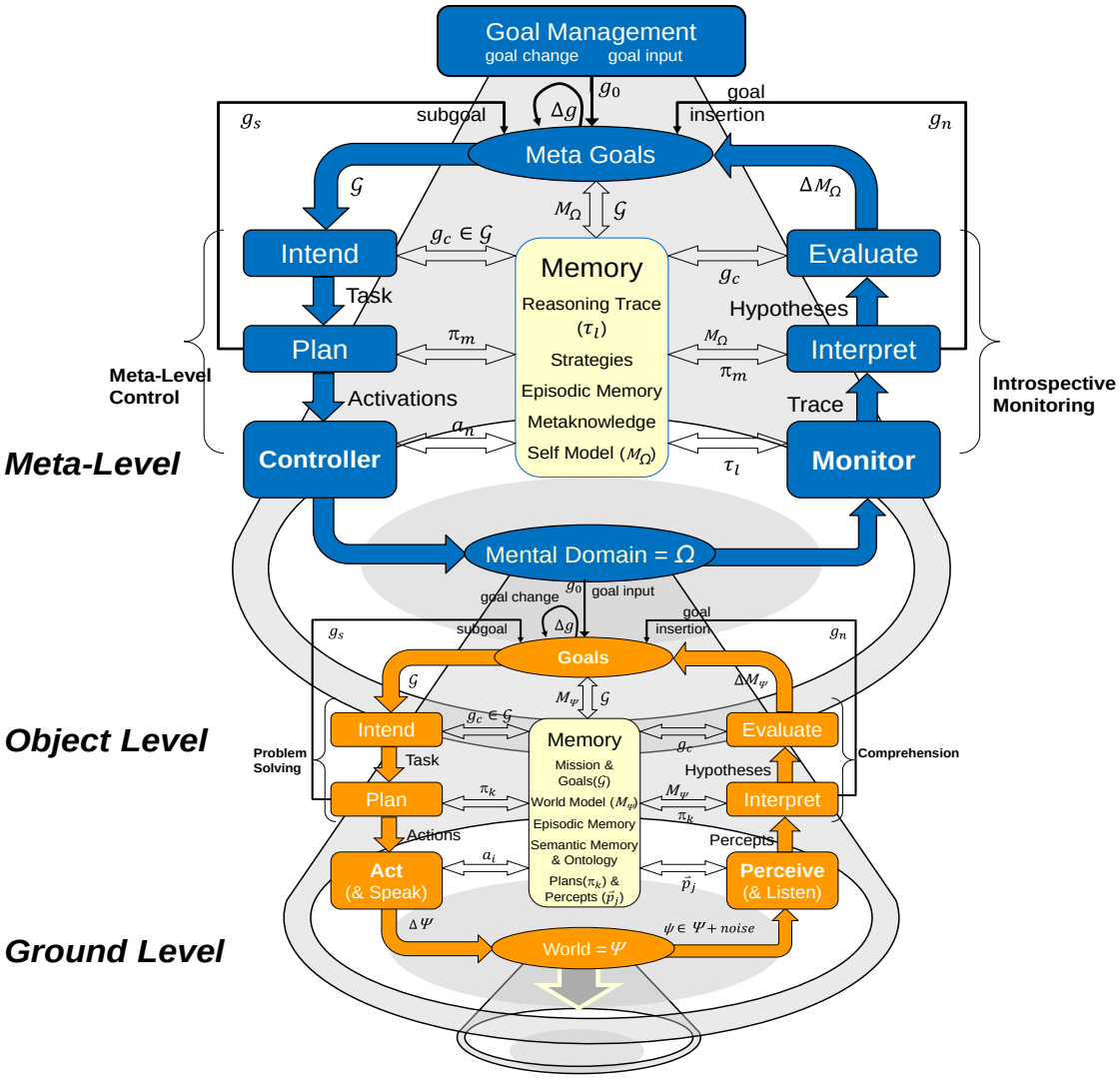

Figure 3. The MIDCA architecture

4. MIDCA

The *Metacognitive, Integrated, Dual-Cycle Architecture (MIDCA)* (Cox, Maynord, Paisner, Perlis, & Oates, 2013; Cox, Oates, & Perlis, 2011) consists of "action-perception" cycles at both the cognitive (i.e., object) level and the metacognitive (i.e., meta-) level (see Figure 3). The output side of each cycle consists of intention, planning, and action execution, whereas the input side consists of perception, interpretation, and goal evaluation. A cycle selects a goal and commits to achieving it. The agent then creates a plan to achieve the goal and subsequently executes the planned actions to make the domain match the goal state. The agent perceives

changes to the environment resulting from the actions, interprets the percepts with respect to the plan, and evaluates the interpretation with respect to the goal. At the object level, the cycle achieves goals that change the environment (i.e., ground level). At the meta-level, the cycle achieves goals that change the object level. That is, the metacognitive "perception" components introspectively monitor the processes and mental state changes at the cognitive level. The "action" component consists of a meta-level controller that mediates reasoning over an abstract representation of the object level cognition.

To appreciate the distinctions in the relationship between levels, examine the finer details of the object level as shown in Figure 3. Here the meta-level executive function manages the goal set $\mathcal{G}$. In this capacity, the meta-level can add initial goals (g_0), subgoals (g_s) or new goals (g_n) to the set, can change goal priorities, or can change a particular goal (Δg). In problem solving, the Intend component commits to a current goal (g_c) from those available by creating an intention to perform some *Task* that can achieve the goal (Cohen & Levesque, 1990). The Plan component then generates a sequence of *Actions* (π_k, e.g., a hierarchical-task-net plan, see Nau, et al., 2003) that instantiates that *Task* given the current model of the world (M_Ψ) and its background knowledge (e.g., semantic memory and ontologies). The plan is executed by the Act component to change the actual world (Ψ) through the effects of the planned *Actions* (a_i). Problem solving stores the goal and plan in memory to provide the agent expectations about how the world will change in the future. Then given these expectations, the comprehension task is to understand the execution of the plan and its interaction with world with respect to the goal so that success occurs.

Comprehension starts with perception of the world in the attentional field via the Perceive component. The Interpret component takes as input the resulting *Percepts* (i.e., $\vec{p}_j$) and the expectations in memory (π_k and g_c) to determine whether the agent is making sufficient progress. A GDA interpretation procedure implements the comprehension process. The procedure is to *note* whether an anomaly has occurred; *assess* potential causes of the anomaly by generating explanatory *Hypotheses*; and *guide* the system through a response. Responses can take various forms, such as (1) test a Hypothesis; (2) ignore and try again; (3) ask for help; or (4) insert another goal (g_n). Otherwise given no anomaly, the Evaluate component incorporates the concepts inferred from the *Percepts* thereby changing the world model (ΔM_Ψ), and the cycle continues. This cycle of problem-solving and action followed by perception and comprehension functions over discrete state and event representations of the environment.

Likewise introspective monitoring starts with "perception" of the self (Ω) via the Monitor component. The Interpret component takes as input the resulting *Trace* (i.e.,τ_l) and the expectations in memory (π_m and g_c) to determine whether the reasoning is making sufficient progress. The Interpret procedure is to *detect* a reasoning failure; *explain* potential causes of the failure by generating explanatory *Hypotheses*; and *generate* a learning goal or attainment goal. Reasoning about the self (e.g., am I knowledgeable about the domain) and the reasoning task enables the agent to determine the difference (i.e., learning vs. attainment goal). If MIDCA produces a learning goal, the meta-level control will create and execute a learning plan to change its knowledge. Attainment goals are passed through to the object level. Given no anomaly, the Evaluate component incorporates the concepts inferred from the *Trace* thereby changing the self-model (ΔM_Ω), and the cycle continues.

4.1 Implementation: MIDCA_1.1

MIDCA_1.1 (Paisner, Maynord, Cox, & Perlis, in press) is a simplified version of the complete MIDCA architecture shown in the schematic of Figure 3. It is currently composed of the cognitive (object level) cycle components shown in Figure 3. The implementation effort has concentrated on a simulator that generates successor states based on valid actions taken in the blocksworld domain; a state interpretation component; and a planner. For the planner, we used SHOP2 (Nau et al., 2003), a domain-independent task decomposition planner. Whereas the full MIDCA architecture has a meta-cognitive component, which manages goals, MIDCA_1.1 has no goal management, and simply passes any new goals from the interpreter directly to the planner. In MIDCA_1.1, the Interpret component consists of an integration of bottom up and top down process as explained below. The Act component is incorporated into the blocksworld simulator, and the Perception component is implicit in the transfer of world state representation to the interpreter.

The GDA interpretation procedure at the object level has two variations that represent a bottom-up, data-driven track and a top-down, knowledge rich, goal-driven track (Cox, Maynord, Paisner, Perlis, & Oates, 2013). The data-driven track we call the D-track; whereas the knowledge rich track we call the K-track. The D-track is implemented by a bottom-up GDA process as follows. A statistical anomaly detector constitutes the first step, a neural network identifies low-level causal attributes of the anomaly, and a machine learning goal classifier provides the goal formulation. The K-track is implemented as a case-based explanation process. The representations for expectations significantly differ between the two tracks. K-track expectations come from explicit knowledge structures such as action models used for planning and ontological conceptual categories used for interpretation. Predicted effects form the expectations in the former and attribute constraints constitute expectation in the latter. D-track expectations are implicit by contrast. Here the implied expectation is that the probabilistic distribution of observations will remain the same. When statistical change occurs instead, an expectation violation is raised.

The D-track interpretation procedure uses a novel approach for noting anomalies. We apply the statistical metric called the A-distance to streams of predicate counts in the perceptual input (Cox, Oates, Paisner, & Perlis, 2012; 2013). This enables MIDCA to detect regions whose statistical distributions of predicates differ from previously observed input. These regions are those where change occurs and potential problems exist.

When a change is detected, its severity and type can be determined by reference to a neural network in which nodes represent categories of normal and anomalous states. This network is generated dynamically with the growing neural gas algorithm (Paisner, Perlis, & Cox, 2013) as the D-track processes perceptual input. This process leverages the results of analysis with A-distance to generate anomaly archetypes, each of which represents the typical member of a set of similar anomalies the system has encountered. When a new state is tagged as anomalous by A-distance, the GNG net associates it with one of these groups and outputs the magnitude, predicate type, and valence of the anomaly.

Goal generation is done through a conjunction of two machine learning algorithms both of which work over symbolic predicate representations of the world (Maynord, Cox, Paisner, & Perlis, 2013). Given a world state interpretation, the state is first classified using a decision tree into one of multiple state classes, where each class has an associated goal generation rule generated during learning. Given an interpretation and a class, different groundings of the

variables of the rule are permuted until either one is found which satisfies that rule (in which case a goal can be generated) or until all permutations of groundings have been attempted (in which case no goal can be generated). This approach to goal insertion is naïve in the sense that it constitutes a mapping between world states and goals which is static with respect to any context.

The K-track GDA procedure is presently implemented with the Meta-AQUA system (Cox & Ram, 1999). In Meta-AQUA frame-based concepts in the semantic ontology provide constraints on expected attributes of observed input and on expected results of planned actions. When the system encounters states or actions that diverge from these expectations, an anomaly occurs. Meta-AQUA then retrieves an explanation-pattern that links the observed anomaly to the reasons and causal relationships associated with anomaly. A goal is then generated from salient antecedents of the instantiated explanation pattern (see also Cox, 2007).

4.2 Firefighting Example: Autonomous goal formulation

To generate goals autonomously, one might statistically train a classifier to recognize a goal given an arbitrary state representation.[4] Maynord, Cox, Paisner, & Perlis (2013) demonstrated the potential of this approach using a knowledge structure called a TF-Tree. As mentioned in the previous section, the TF-Tree combines the results of two machine learning algorithms to detect those conditions that warrant the generation of a new goal. Given multiple examples of state-goal pairs, the classifier learns to generate appropriate goals when presented with novel states.

For the implementation of MIDCA_1.1 we used a modified blocksworld for the domain. This version of blocksworld includes both rectangular and triangular blocks that compose the materials for simplified housing construction (see Figure 1). The initial goals for problems in this domain are to build houses consisting of towers of blocks with a roof on each. In addition the possibility exists that blocks may catch fire (set by a hidden arsonist). Furthermore there are additional actions added to the standard blocksworld operators. One action will put out fires, and another will find and capture the arsonist. In the amalgamated firefighting/house-construction/blocksworld domain, the TF-Trees will learn to generate a goal to have a fire extinguished when given a state containing a block on fire.

The fires are problems because of the effect on housing construction and the supposed profits of the housing industry, and they pose threats to life and property. The approach to understanding the fire problems is to ask *why* the fires were started and not just *how*. An explanation of how the fire started would relate the presence of sufficient heat, fuel, and oxygen with the combustion of the blocks. Generating the negation of the presence of the oxygen for example would result in the goal ¬oxygen and therefore put out the fire. But this does not get to the reason the fire started in the first place. To ask why the fire was started would result in possibly two hypotheses or explanations. Poor safety conditions can lead to fire or the actions of arsonists can result in fire. In this latter case, the arsonist causes the presence of the heat through some hidden lighting action. Given this explanation the agent can anticipate the threat of more fires and generate a goal

[4] One might also just enumerate all possible goals and the conditions under which they are triggered. Tac-Air Soar (Jones, Laird, Tambe, & Rosenbloom, 1994) takes such an approach. Operators exist for various goal types and data-driven context-sensitive rules spawn them given matching run-time observations. However even if one could engineer all relevant goals for a domain and all the conditions under which they apply, an expectation failure or surprise may occur if the domain shifts (e.g., the introduction of novel technology). The recognition of new problems and the explanation of their causes would enable the formulation of a goal from first principles, even under conditions not envisioned by the agent designer.

to remove the threat by finding the arsonist. Apprehending the arsonist then removes the potential of fires in the future rather than just reacting to fires that started in the past.

Empirical results show that the GDA approach to goal formulation significantly outperforms the statistical approach with fewer actions required for the same amount of housing construction. In brief the housing domain goes through a cycle of three state classes in building new "houses." Figure 4 shows three instantiated states classified by the TF-Trees and the grounded goals that each tree recognizes.

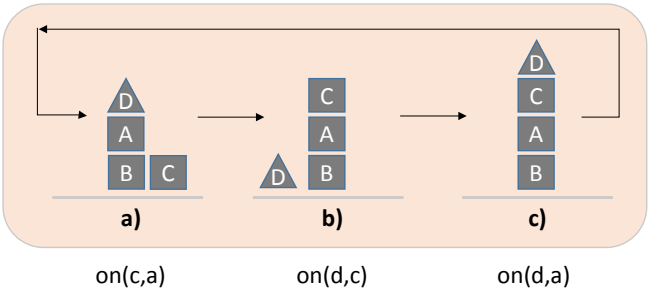

Figure 4. Three classifiers exist that recognize goals to get to the next state

Using the GDA method, over 1000 time steps results in an average improvement of 245.4% relative to a baseline method that lacks goal generation altogether; whereas the statistical method results in a 54.2% improvement over the baseline (see Paisner, Maynord, Cox, & Perlis, in press for technical details). The specific results are less important, however, than the conceptual framing of the problem. Many methods exist to implement a GDA approach to planning and action (e.g., see Aha, Klenk, Muñoz-Avila, Ram, & Shapiro, 2010; Aha, Cox, & Munoz-Avila, in press). The crucial point for autonomous cognitive systems is to consider how they can be made to understand and represent the problem not just optimize a solution.

5. Conclusion

This paper is not a technical analysis of autonomy, nor is it an empirical investigation of specific algorithms and data structures; rather it makes a sometimes intuitive case for a different approach to autonomy and intelligent agency. The proposal is that autonomy is chiefly about recognizing problems independently and generating goals to solve them. The specific technical details as they exist now are to be found in the references cited throughout this article. In lieu of these details here, I have tried instead to explore an alternative that lies relatively unexamined and to shed some light on the issues. For the most part, my proposition is novel, because few have thought much about it computationally; the research that does exists is found mainly in the social psychology literature. Also it is still unclear how many of the propositions and claims herein can be implemented fully, although further examples exist, especially in some of the earlier AI literature (e.g., Schank, 1986; Ram, 1991). A few new researchers have done work under the GDA topic, but much of the focus has been on goal formulation, explanation, and case learning rather than problem recognition and question posing. So as it stands now, much research remains, not the least of which is to clarify the role of question asking and experience in problem recognition.

No current theory explains precisely why humans question the world and themselves. What is the exact relationship between question posing, problem recognition, and motivation in an intelligent agent? Why does an agent actively look for problems to begin with? That is why are we so motivated (sometimes compelled) to solve problems? Certainly a benefit accrues or some personal utility exists in the cost-benefit calculations that percolate in our mind, but this may not

constitute the whole story. Perhaps at least part of the answer can be seen in the explanations provided by Higgins (2012).

Higgins proposes that human intelligence is motivated by three things: value, truth, and fit. Value corresponds to the utility (i.e., value) we experience in things (e.g., objects, events, or states). Truth corresponds to the capability to accurately interpret the world with respect to one's own experience and memory. Fit refers to the appropriateness of strategy. The AI community focusses on the first factor; whereas this paper has focused on the second. The motivation of truth impels us to look clearly at the world. We seek to discern where our observations agree with and where they differ from our knowledge of the world with the aim of improving our understanding. As I see it, value and truth correspond to the problem solving and problem comprehension divisions within the MIDCA architecture. An autonomous agent values effective performance and finds "truth" in effective interpretation. One is as a complement to the other.

Acknowledgements

This material is based upon work supported by ONR Grants # N00014-12-1-0430 and # N00014-12-1-0172 and by ARO Grant # W911NF-12-1-0471. I thank Michael Maynord, Tim Oates, Don Perlis and the anonymous reviewers for comments on the content of this draft.

References

Aha, D. W., Cox, M. T., & Munoz-Avila, H. (Eds.) (in press). *Proceedings of the 2013 Annual Conference on Advances in Cognitive Systems: Workshop on Goal Reasoning*. College Park, MD: University of Maryland.

Aha, D. W., Klenk, M., Muñoz-Avila, H., Ram, A., & Shapiro, D. (Eds.) (2010). Goal-Directed Autonomy: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press.

Anderson, M., & Perlis, D. (2005). Logic, self-awareness and self-improvement. *Journal of Logic and Computation 15*, 21–40.

Berry, A. J., Howitt, J., Gu, D.-W., & Postlethwaite, I. (2012). A continuous local motion planning framework for unmanned vehicles in complex environments. *Journal of Intelligent & Robotic Systems 66*(4), 477-494.

Birnbaum, L., Collins, G., Freed, M., & Krulwich, B. (1990). Model-based diagnosis of planning failures. In *Proceedings of the Eighth National Conference on Artificial Intelligence* (pp. 318-323). Menlo Park, CA: AAAI Press.

Cohen, P. R. & Levesque, H. J. (1990). Intention is choice with commitment. *Artificial Intelligence 42*(2-3), 213 – 261.

Cox, M. T. (2007). Perpetual self-aware cognitive agents. *AI Magazine 28*(1), 32-45.

Cox, M. T. (2011). Metareasoning, monitoring, and self-explanation. In M. T. Cox & A. Raja (Eds.) *Metareasoning: Thinking about thinking* (pp. 131-149). Cambridge, MA: MIT Press.

Cox, M. T., Maynord, M., Paisner, M., Perlis, D., & Oates, T. (2013). The integration of cognitive and metacognitive processes with data-driven and knowledge-rich structures. In *Proceedings of the Annual Meeting of the International Association for Computing and Philosophy*.

Cox, M. T., Oates, T., Paisner, M., & Perlis, D. (2012). Noting anomalies in streams of symbolic predicates using A-distance. *Advances in Cognitive Systems 2*, 167-184.

Cox, M. T., Oates, T., Paisner, M., & Perlis, D. (2013). Detecting change in diverse symbolic worlds. In L. Correia, L. P. Reis, L. M. Gomes, H. Guerra, & P. Cardoso (Eds.), *Advances in Artificial Intelligence, 16th Portuguese Conference on Artificial Intelligence* (pp. 179-190). University of the Azores, Portugal: CMATI.

Cox, M. T., Oates, T., & Perlis, D. (2011). Toward an integrated metacognitive architecture. In P. Langley (Ed.), *Advances in Cognitive Systems: Papers from the 2011 AAAI Fall Symposium* (pp. 74-81). Technical Report FS-11-01. Menlo Park, CA: AAAI Press.

Cox, M. T., & Ram, A. (1999). Introspective multistrategy learning: On the construction of learning strategies. *Artificial Intelligence, 112*, 1-55.

Cox, M. T., & Veloso, M. M. (1998). Goal transformations in continuous planning. In M. desJardins (Ed.), *Proceedings of the 1998 AAAI Fall Symposium on Distributed Continual Planning* (pp. 23-30). Menlo Park, CA: AAAI Press.

Defense Advanced Research Projects Agency (2012). *ASW Continuous Trail Unmanned Vessel (ACTUV) Phases 2 through 4*. DARPA-BAA-12-19. Arlington, VA: DARPA. https://www.fbo.gov/utils/view?id=2935bca24073347c8fd1ae0820cc20f8

Duda, R. O., & Shortliffe, E. H. (1983). Expert systems research. *Science 220*, 261-268.

Dweck, C. S. (1986). Motivational processes affecting learning. *American Psychologist, 41*, 1040-1048.

Franklin, S., & Graesser, A. (1997) Is it an agent, or just a program?: A taxonomy for autonomous agents, *Intelligent agents III*. Berlin: Springer, 21-35.

Ghallab, M., Nau, D., & Traverso, P. (2004). *Automated planning: Theory and practice*. San Francisco: Morgan Kaufmann.

Getzels, J. W., & Csikszentmihalyi, M. (1975). From problem solving to problem finding. In I. A. Taylor & J. W. Getzels (Eds.), *Perspectives in creativity* (pp. 90-116). Chicago: Aldine.

Getzels, J. W. (1979). Problem finding: A theoretical note. *Cognitive Science 3*, 167-172.

Hagen, P. E., Midtgaard, O., & Hasvold, O. (2007). Making AUVs truly autonomous. In *Proceedings of the MTS/IEEE Oceans Conference and Exhibition* (pp. 1-4). Red Hook, NY: Curran Associates.

Hanheide, M., Hawes, N., Wyatt, J., Gobelbecker, M., Brenner, M., Sjoo, K., Aydemir, A., Jenselt, P., Zender, H. and Kruiff, G. (2010). A framework for goal generation and management. In D. W. Aha, M. Klenk, H. Muñoz-Avila, A. Ram, & D. Shapiro (Eds.), *Goal Directed Autonomy: Papers from the AAAI Workshop*. Menlo Park, CA: AAAI Press.

Hawkins, D. I., Best, R. J., & Coney, K. A. (1989). *Consumer behavior: Implications for marketing strategy, 4ed*. Boston: Business Publications.

Higgins, E. T. (2012). *Beyond pleasure and pain: How motivation works*. New York: Oxford University Press.

Holland, J. H. (1986). Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In R. Michalski, J. Carbonell & T. Mitchell (Eds.), *Machine learning: An artificial intelligence approach, Vol. 2* (pp. 593-623). San Mateo, CA: Morgan Kaufmann Publishers.

Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P., & Koss, F. V. (1999). Automated intelligent pilots for combat flight simulation. *AI Magazine, 20*(1), 27-41.

Klein, G., Pliske, R. Crandall, B., & Woods, D. D. (2005). Problem detection. *Cognition, Technology & Work 7*(1), 14-28.

Klenk, M., Molineaux, M., & Aha, D. (2013). Goal-driven autonomy for responding to unexpected events in strategy simulations. *Computational Intelligence, 29*(2), 187–206, 2013.

Kruglanski, A. W. (1996). Goals as knowledge structures. In P. M. Gollwitzer & J. A. Bargh (Eds.), *The psychology of action: Linking cognition and motivation to behavior* (pp. 599-618). New York: Guilford Press.

Kruglanski, A. W., Köpetz, C., Bélanger, J. J., Chun, W. Y., Orehek, E., & Fishbach, A. (2013). Features of multifinality. *Personality and Social Psychology Review, 17*(1) 22–39.

Kruglanski, A. W., Shah, J. Y., Fishbach, A., Friedman, R., Young, W., & Chun (2002). A theory of goal systems. In M. P. Zanna (Ed.), *Advances in experimental social psychology* (pp. 331-378). New York: Academic Press.

Laird, J. E. (2012). *The Soar cognitive architecture*. Cambridge, MA: MIT Press.

Laird, J. E., Derbinsky, N., & Tinkerhess, M. (2012). Online determination of value-function structure and action-value estimates for reinforcement learning in a cognitive architecture. *Advances in Cognitive Systems 2*, 221-238.

Laird, J., Rosenbloom, P., & Newell, A. (1986). *Universal subgoaling and chunking: The automatic generation and learning of goal hierarchies*. Norwell, MA: Kluwer Academic.

Lenat, D., & Guha, R. (1989). *Building large knowledge-based systems*. Menlo Park, CA: Addison-Wesley.

Li, N., Stracuzzi, D. J., & Langley, P. (2012). Improving acquisition of teleoreactive logic programs through representation extension. *Advances in Cognitive Systems 1*, 109–126.

Maes, P. (1994). Modeling adaptive autonomous agents. *Artificial Life 1* (1-2), 135-162.

Malle, B. F. (2004). *How the mind explains behavior: Folk explanations, meaning, and social interaction*. Cambridge, MA: MIT Press/Bradford Books.

Maynord, M., Cox, M. T., Paisner, M., & Perlis, D. (2013). Data-driven goal generation for integrated cognitive systems. In C. Lebiere & P. S. Rosenbloom (Eds.), *Integrated Cognition: Papers from the 2013 Fall Symposium* (pp. 47-54). Menlo Park, CA: AAAI Press.

Minguez, J., Lamiraux, F., & Laumond, J.-P. (2008). Motion planning and obstacle avoidance. In B. Siciliano & O. Khatib (Eds.), *Springer handbook of robotics* (pp. 827-852). Berlin: Springer.

Munoz-Avila, H., Jaidee, U., Aha, D. W., Carter, E. (2010). Goal-driven autonomy with case-based reasoning. In *Case-Based Reasoning. Research and Development, 18th International Conference on Case-Based Reasoning*, ICCBR 2010 (pp. 228-241). Berlin: Springer.

Nau, D., Au, T., Ilghami, O., Kuter, U., Murdock, J., Wu, D., & Yaman, F. (2003). SHOP2: An HTN planning system. *Journal of Artificial Intelligence Research 20*, 379–404

Norman, T. (1995). *Motivation-based direction of planning attention in agents with goal autonomy*. PhD thesis, Department of Computer Science, University College London.

Paisner, M., Maynord, M., Cox, M. T., & Perlis, D. (in press). Goal-driven autonomy in dynamic environments. To appear in D. W. Aha, M. T. Cox, & H. Munoz-Avila (Eds.), *Proceedings of*

the 2013 Annual Conference on Advances in Cognitive Systems: Workshop on Goal Reasoning. College Park, MD: University of Maryland.

Paisner, M., Perlis, D., & Cox, M. T. (2013). Symbolic anomaly detection and assessment using growing neural gas. In *Proceedings of the 25th IEEE International Conference on Tools with Artificial Intelligence* (pp. 175-181). Los Alamitos, CA: IEEE Computer Society.

Perlis, D. (2011). There's no 'me' in meta - or is there? In Cox, M. T., and Raja, A., (Eds.), *Metareasoning: Thinking about thinking* (pp. 15–26). Cambridge, MA: MIT Press.

Pretz, J. E., Naples, A. J., & Sternberg, R. J. (2003). Recognizing, defining, and representing problems. In J. E. D. a. R. J. Sternberg (Ed.), *The psychology of problem solving* (pp. 3-30). Cambridge, UK: Cambridge University Press.

Ram, A. (1990). Decision models: A theory of volitional explanation. In *Proceedings of Twelfth Annual Conference of the Cognitive Science Society* (pp. 198-205). Hillsdale, NJ: LEA.

Ram, A. (1991). A theory of questions and question asking. *Journal of the Learning Sciences*, *1*, (3&4), 273-318.

Ram, A., & Leake, D. (1995). Learning, goals, and learning goals. In A. Ram & D. Leake (Eds.), *Goal-driven learning* (pp. 1-37). Cambridge, MA: MIT Press/Bradford Books.

Rilke, R. M. (1986). *Letters to a young poet*. New York: Vintage Books. Originally published 1903.

Runco, M. A., & Chand, I. (1994). Problem finding, evaluative thinking, and creativity. In M. A. Runco (Ed.), *Problem finding, problem solving, and creativity* (pp. 40-76). Norwood, NJ: Ablex.

Russell, S. & Norvig, P. (1995). *Artificial intelligence: A modern approach*. Upper Saddle River, NJ: Prentice Hall.

Schank, R. C. (1982). *Dynamic memory: A theory of reminding and learning in computers and people*. Cambridge, MA: Cambridge University Press.

Schank, R. C. (1986). *Explanation patterns: Understanding mechanically and creatively*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R. C. (1994). Goal-based scenarios: A radical look at education. *The Journal of the Learning Sciences, 3*(4), 429-453.

Schank, R. C., & Abelson, R. P. (1977). *Scripts, plans, goals and understanding: An inquiry into human knowledge structures*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R. C., & Owens, C. C. (1987). Understanding by explaining expectation failures. In R. G. Reilly (Ed.), *Communication failure in dialogue and discourse*. New York: Elsevier Science.

Stone, P. & Veloso, M. M. (2000). Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots. 8*, 345-383.

Talamadupula, K., Schermerhorn, P., Benton, J., Kambhampati, S., & Scheutz, M. (2011). Planning for agents with changing goals. In *Proceedings of the 21st International Conference on Automated Planning and Scheduling* (pp. 71-74). Menlo Park, CA: AAAI Press.

Vattam, S., Klenk, M., Molineaux, M., & Aha, D. (in press). Breadth of approaches to goal reasoning: A research survey. To appear in D. W. Aha, M. T. Cox, & H. Munoz-Avila (Eds.), *Proceedings of the 2013 Annual Conference on Advances in Cognitive Systems: Workshop on Goal Reasoning*. College Park, MD: University of Maryland.

Veloso, M. (1994). *Planning and learning by analogical reasoning*. Berlin: Springer-Verlag.

Weber, B. G., Mateas, M., & Jhala, A. (2010). Case-based goal formulation. In *Proceedings of the AAAI Workshop on Goal-Driven Autonomy*.

Weiss, G. (1999). *Multiagent systems: A modern approach to distributed artificial intelligence*. Cambridge, MA: MIT Press.

Wilson, M., Molineaux, M., & Aha, D. W. (2013). Domain-independent heuristics for goal formulation. *Proceedings of the Twenty-Sixth Florida Artificial Intelligence Research Society Conference* (pp. 160-165). Menlo Park, CA: AAAI Press.

Wooldridge, M. (2002). *An introduction to multiagent systems*. Hoboken, NJ: Wiley.