Cognitive Automation Solves Many AI-Hard Problems

Russell R. Vane III Douglas Griffith

General Dynamics Advanced Information Systems
1400 Key Blvd., Suite 1200
Arlington, VA 22209
russ.vane@gd-ais.com douglas.grffith@gd-ais.com

Abstract

Cognitive Automata (or CogBots) address four of the challenge questions of the Between a Rock and a Hard Place Symposium in the affirmative. This approach was designed in 2003, primitively implemented in 2004 and submitted for patent in 2005. A CogBot performs "good enough" calculations and searches; in a multi-level, heterogeneous network; adapting to subtle contexts; and using specialized components (concept automata) to exhibit general intelligence.

Introduction

The difficulty of AI problems is that they have not been difficult enough to eschew brute force or domain specific solutions. AI researchers are in a comfortable, fairly well-paid intellectual cul-de-sac. Cognitive scientists began their research without the neurophysiology tools of today [Restak 2003] so they are in an (often less well-paid) experimental cul-de-sac, not unlike much of experimental psychology. What is needed is a multi-disciplinary approach to cognitive automation that preserves the original goals of both disciplines. This paper addresses four of the workshop questions positively. We have found that "good enough" solutions (1) are valuable when controlled by a multi-level heterogeneous approach (2) that adapts to subtle contextual changes both locally (within level) and globally (within the network) (5) while asking specialized, modular components, called conceptual automata (cobots), to exhibit general intelligence(6) and accomplish the mission.

Background

J.C.R. Licklider [Licklider 1960] was a true visionary. He realized early on that there was an exciting potential for man-computer symbiosis. This approach has been updated by Griffith [Griffith 2005] as neosymbiosis. Griffith, a cognitive psychologist, and Vane have also addressed the idea that humans should be included in systems and even "algorithms" to increase their robustness. Such neosymbiotic systems would actually appear to observers as intelligent augmentation (IA) in contrast to pure AI.

Herb Simon [Simon 1947] still has a lot to say about cognitive automation. Humans in organizations often satisfice, by doing "good enough" plans that deliver "good enough" results. They have bosses with missions, people with skills and tasks, and an interlingua for updating each other. So should cognitive automata (we call CogBots). Emulation of such organizations is a guiding principle of GDAIS's cognitive automation.

DARPA's Cougaar program (1997-2004) delivered a valuable agent framework that is being extended by the open source community and Cougaar Software, Inc.'s Active Edge™. No further discussion of agent frameworks occurs in this paper, except to say that CogBots are built on it. It provides the interruptible, multi-threaded features that are needed by dynamic, flexible cognitive automata.

Discussion

James Albus [Albus 1988] had an insightful realization when moving from factory automation to autonomous robots – they need perception to transform information into models and they need judgment to add meaning to perception and prediction. Cognitive automata need sensory processing and judgment (see figure 1 on the next page).

This architecture supports the observe-orient-decide-act (OODA) loop invented by John Boyd [Boyd 1987], as updated and expanded as a bio-informatics paradigm by John Bodnar [Bodnar 2003]. There are many similarities with W. Edwards Deming's popularizing of Shewhart's Plan-Do-Check-Act loop [Shewhart 1939] that personifies the quality assurance activities of highly efficient organizations. This is both a meta-strategy and a conceptual tactic.

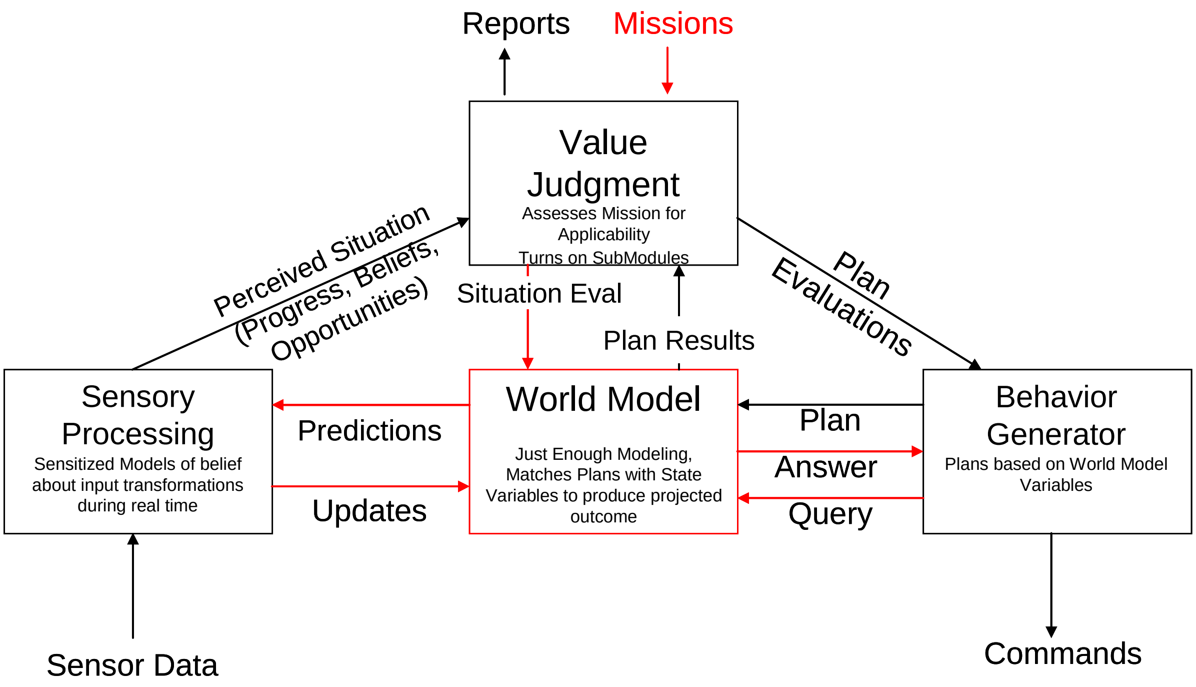
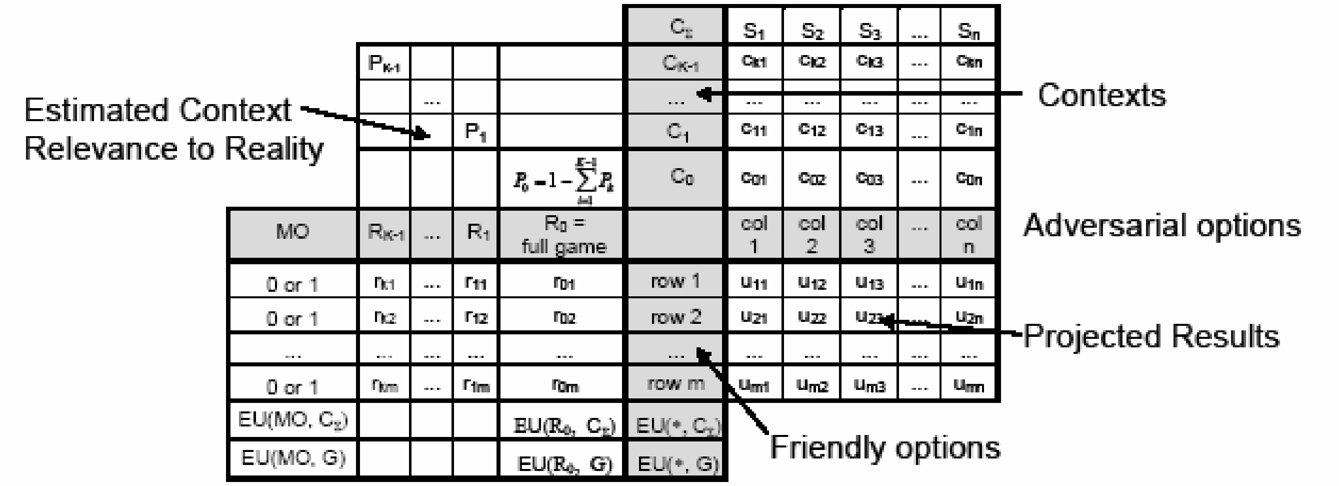

Reports
Missions
Value Judgment
Assesses Mission for Applicability
Turns on SubModules
Perceived Situation (Progress, Beliefs, Opportunities)
Situation Eval
Plan Results
Plan Evaluations
Sensory Processing
Sensitized Models of belief about input transformations during real time
Predictions
Updates
World Model
Just Enough Modeling, Matches Plans with State Variables to produce projected outcome
Plan
Answer
Query
Behavior Generator
Plans based on World Model Variables
Sensor Data
Commands

Figure 1. Albus's Real-time Control System-4, Updated as a Concept Automaton


Estimated Context Relevance to Reality
Contexts
Adversarial options
Projected Results
Friendly options

					C_Σ	S_1	S_2	S_3	...	S_n
	P_{K-1}				C_{K-1}	c_{K1}	c_{K2}	c_{K3}	...	c_{Kn}
		...			...	...	...	...	...	...
			P_1		C_1	c_{11}	c_{12}	c_{13}	...	c_{1n}
				$P_0 = 1 - \sum_{k=1}^{K-1} P_k$	C_0	c_{01}	c_{02}	c_{03}	...	c_{0n}
MO	R_{K-1}	...	R_1	R_0 = full game		col 1	col 2	col 3	...	col n
0 or 1	r_{K1}	...	r_{11}	r_{01}	row 1	u_{11}	u_{12}	u_{13}	...	u_{1n}
0 or 1	r_{K2}	...	r_{12}	r_{02}	row 2	u_{21}	u_{22}	u_{23}	...	u_{2n}
...	...	...	...	...	...	...	...	...	...	...
0 or 1	r_{Km}	...	r_{1m}	r_{0m}	row m	u_{m1}	u_{m2}	u_{m3}	...	u_{mn}
EU(MO, C_Σ)				EU(R_0, C_Σ)	EU(*, C_Σ)					
EU(MO, G)				EU(R_0, G)	EU(*, G)					

Figure 2. Hypergame theory normal form

Expert human judgment is not well emulated by decision theory and game theory, particularly when considering its dynamic aspects. Decision theory's weaknesses are its penchant to ignore any option that has an estimated zero probability of happening. Such information provides circumspective evidence for estimating risk or the effects of uncertainty. Whereas game theory assumes consistent alignment of beliefs by knowing the values that other players place on their options. This inconsistency assumes that players know every option of everyone else and how they value it, yet insists that no one can build a model of any player's intent. These dichotomies were directly addressed by hypergame theory in figure 2. The concept is that all players are playing different conceptual games regardless of the shared physical game reality. Peter G. Bennett's [Bennett, Dando, 1979] approach allows hypergame theory to address perceptual, evidential, contextual reasoning. Vane invented the hypergame normal form [Vane 2000] and addressed agent augmentation [Vane, Griffith 2005]. Judgment is needed to assess confusion, monitor progress, and assess the value of activities in the support of the mission of cognitive automata.

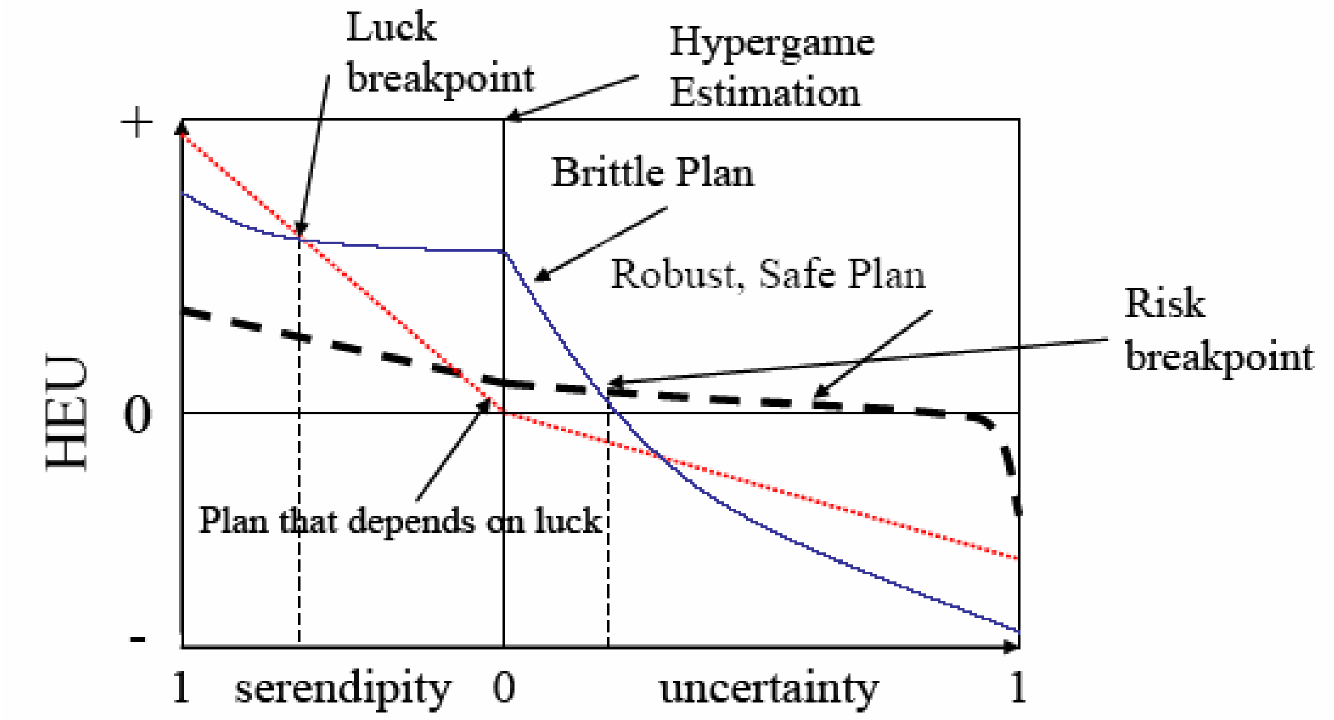


Figure 3. Hypergame Expected Utility (HEU) considers uncertainty and serendipity

In figure 3, the y-axis is the value or utility to a player for hypergame conclusions. This figure shows brittle high-value plans are often not the right choice when uncertainty increases.

Keith Devlin [Devlin] has some important insights into context as a problem statement and problem solution influencer. The formula below means that the context of the owner is necessarily a superset of what is thought about the specific situations potentially influencing a task being performed. All of these influence the model selected, Model, with m factors, f, to assess the truthfulness of informational sentence, si. Models can be factored and implemented in software, contexts are never completely factorable.

$$C_{owner} \triangleright C_{owner,situations} \triangleright C_{task} \models Model[f_1, f_2, f_3 ... f_m] \models s_i$$

A cardinal principle is to require context to remain external to the model. It is recognized that context influences the results or interpretation of the results. Also any context in the context chain separated from the model by the “|=” can be used as a standalone interpreter of the model. That is the context of the ‘owner’ can have one interpretation of the model’s validity, as can the context of the task. But bringing context inside the model changes the context and the model. It is human wisdom (so far) that decides the appropriateness of the parsimony of any model. Reasoning about the impact of moving an aspect of the context into the model or removing a component of a model can be made using hypergame theory.

CogBots maintain a tree of missions being addressed by any concept automaton (cobot) similar to a calling stack in computer science that represents the CogBot’s understanding of context. There are many items of embedded knowledge and value affecting why these missions were chosen. These must be isolated from the subordinate cobot for efficiency purposes. Yet we have or will have most of this information available in software. It is an interesting research problem as to whether or how often we create new cobots that are hybrids of a specific cobot based on what will surely become recognizable patterns of influence (essentially a generality of analogical reasoning).

Cognitive scientists have to become the computer scientist’s conscience here. Let’s agree that abstract principles and related heuristics are important. Value judgment is important, as it is in humans. Whenever we are tempted to transform any research problem into a model albeit huge via linear programming, simulation, a swarm of agents, and so on; we need to be drawn up short. We can never do this in anything but toys or games. We truly need a cognitive approach to these issues to scale to the real world.

Instead of bigger models, we need "trusted" ones. By authoring concept automata (cobots) that are simple enough that they can be verified by continual reuse, critical thinking or code review; we are providing the building blocks of reusable knowledge. This is the reason that we teach children mathematics, physics, and health sciences. We are giving them hard-earned knowledge vicariously to increase their cognitive reach. We need to do the same for CogBots. Devlin has hypothesized and begun to research trust protocols.

A concrete example of such a context hierarchy is shown in figure 4. The CogBot is managing the tree by its beliefs about information and judgment about the implied expected results of embedded cobot strategies. Our concept automata are the embodiment of a context management system that allows conceptual knowledge to be interpreted properly.

In figure four the unselected options are next to a cobot rectangle, whereas the selected one appears as the order to the next cobot. The context is relieve thirst by getting a drink where easier is preferred by cobot one, in the refrigerator by cobot two, and preserve my diversity of selections is cobot three. In this case the "why not your favorite drink?" question could be answered "I wanted it quickly" rather than "Because I wanted the most plentiful one," since that is rarely knowable or even sensible in contexts other than an open refrigerator. Or even a more general "why?" could be answered "I was thirsty." Whereas, "why not water?" is answered "The refrigerator is closer."

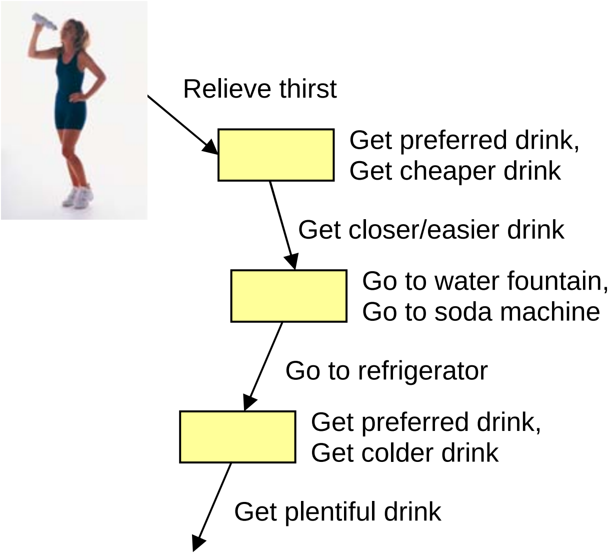


Figure 4. Labeled context tree in a CogBot (network of cobots)

Findings

Cobots estimate when they are confused using their applicability function in the judgment component based on what is being observed in their believe component. This is a critical subsystem, isolating the interlingua in the network from the needs of the model. This transduction is preconditioned by what the model expects, thus surprise can actually be observed and reported.

Humans will need to trust a system to actually use it in real world situations. This requires a large degree of transparency already aided by the computer recognizing when it is confused and reporting it appropriately. The most important aspect however is a cognitive automaton's capacity to report "why" it did something. By tracking the multiple hypotheses and contexts that alternative strategies might have yielded, a CogBot network recognizes which new strategy would change the context the most. This partial of utility with respect to a change in context is the reason that humans give when they are asked "why." Now we have a similar mechanism in a CogBot.

$$why = \max\left(\frac{\partial U_i}{\partial C_i}\right)_{i=1}^{N} \approx \max\left(\left|U_{chosen} - U_i\right| \cdot p(C_{chosen}, C_i)\right)_{i=1}^{N}$$

Each context is not specified. Thus, when one says that a partial exists over N different contexts, we're talking about the CogBot's model of near contexts that were not chosen and the likelihood that they might be active. Furthermore the magnitude of the change in utility is important to discern significance from a competing option. For example, you chose to walk away from the roulette table. Why? Because you did not want to watch people lose money (a difference in utility) and it was very likely that the gambling casino would continue to win (probability was high). It could have equally well been that you had an appointment with a friend (also a highly probable change in utility, but positive this time). This can be further refined by considering the context(s) of the question asker.

Conclusions

By knowing "why" CogBots also provide their developers with a lot of information about the benefits of new cobots, better embedded knowledge, the benefits of handling new/more information. As more experience is gained, new "cognitive" learning mechanisms will be able to be proposed by cognitive scientists. One of these my be the hybridization discussed between context and model. Another example using a history of the last n times that a cobot is called to refine its model when idle (a sort of REM sleep for the cobot), where n is determined per cobot cognitively and experimentally.

One consequence is that such systems could be built incrementally, similar to Gilb's [Gilb 2005] evolutionary delivery using quantified requirements, today. Then cognitive scientists and computer scientists could decide together what the next delivery should be regardless of whether it is AI or not. New models, better sensing, better distribution on multiple platforms, and so on become meaningful because they are useful – not because we are trying to get the system to be humanless.

A CogBot, a network of cobots, represents an encyclopedia of encapsulated knowledge modules, each more or less useful depending on the dynamics of context whether supplied internally or externally by interacting with reality. This really supplants prominent ideas of ontology and knowledge interchange languages.

Cognitive scientists, applied mathematicians, and AI researchers need to embrace issues of trust, model transparency, context, meaning, value judgments, creativity, cognitive limits, and cognitive biases using Gilb's quantified requirements. We need the right goals to be addressed by the entire community.

References

Albus, J. 1993. "A Reference Model Architecture for Intelligent Systems Design, NIST

Bennett, P.G., Dando, M.R., 1979. "Complex Strategic Analysis: A Hypergame Study of the Fall of France," *Journal of the Operational Research Society* **30**: 23-32

Bodnar, J. W. 2003. *Warning Analysis for the Information Age: Rethinking the Intelligence Process*, Joint Military Intelligence College, Washington, DC

Boyd, J. R. (Col). 1987. *A Discourse on Winning and Losing.* Collection of un-numbered briefing slides. Aug 1987 More can be found on Boyd at http://www.d-n-i.net; in Robert Coram, *Boyd: The Fighter Pilot Who Changed the Art of War*, Boston: Little, Brown and Co., 2002, and in Grant Hammond, *The Mind of War: John Boyd and American Security*, Washington, D. C.: Smithsonian Institution Press, April 2001

Devlin, K. M. 2004. "Confronting context effects in intelligence analysis: can mathematics really help?" http://www.stanford.edu/~kdevlin/Context_in_Reasoning.pdf (53 pp. in-progress)

Gilb, T. 2005. *Competitive Engineering*. Elsevier Butterworth Heineman, 474 pp.

Griffith, D. 2005. "Neo-symbiosis: a tool for diversity and enrichment." CybErg 2005, cyberg.wits.ac.za

Licklider, J.C.R. (1960). Man-computer symbiosis. IRE Transactions on Human Factors in Electronics., HFE ,4-11.

Restak, R. 2003. *The New Brain*. Rodale Books, 228 pp.

Shewhart, W. 1939. *Statistical Method from the Viewpoint of Quality Control*, The Graduate School of Agriculture, Washington

Simon, H.A., 1947. *Administrative Behavior: A study of decision-making processes in administrative organization*. Macmillan, New York, (4th edition, 1997, The Free Press)

Vane, R.R. 2000. *Using Hypergame Theory to Select Plans in Competitive Situations*, Ph.D. Dissertation, George Mason University

Vane, R.R., Griffith, D. 2005. "Augmenting Cognition for Adversarial Reasoning," *First International Augmented Cognition Conference 2005*, Las Vegas, NV