2009-08

Evolutionary Adaptive Development: A New Foundation for Creating Intelligent Complex Systems

Hayes-Roth, F.

Hayes-Roth, F. (2009). Evolutionary Adaptive Development: A New Foundation for Creating Intelligent Complex Systems. Singularity University, NASA Ames, Mountain View, CA. August 1, 2009.
https://hdl.handle.net/10945/37756

Evolutionary Adaptive Development:

A New Foundation for Creating Intelligent Complex Systems

Rick Hayes-Roth, Professor, Information Sciences, Naval Postgraduate School
hayes-roth@nps.edu

Workshop for Singularity University
August 1, 2009

13 Key Ideas

1. AI as a programming exercise isn't scalable

The engineering of AI products & systems has been climbing a local, non-global optimum with relatively diminishing returns

2. Vastly more powerful AI products require automated methods

The machines need to be able to discover, experiment, adapt and evolve

They need to run vast numbers of "experiments"

They need to be endowed initially by programmers with these capabilities

3. Learning means improving one's capability for effective behavior

Efficient Thought provides a framework for understanding those terms

Behaviors produce outcomes in actual operational contexts

Outcomes and events that differ from expectations provide the stimulus for most significant learning

Outcomes that mostly accord with expectations confirm behaviors and strengthen their constituents

4. Models enable and underlie Efficient Thought and are the objects of learning

There are many types and domains for models

Much of human science and engineering focuses on model building and tuning

To date, machines have incorporated and used few models

Parameter identification (modeling fitting), strengthening and tuning by feedback mechanisms is pretty standard today

Model construction by machines is very primitive today

Model construction by human-machine teams is the leading edge

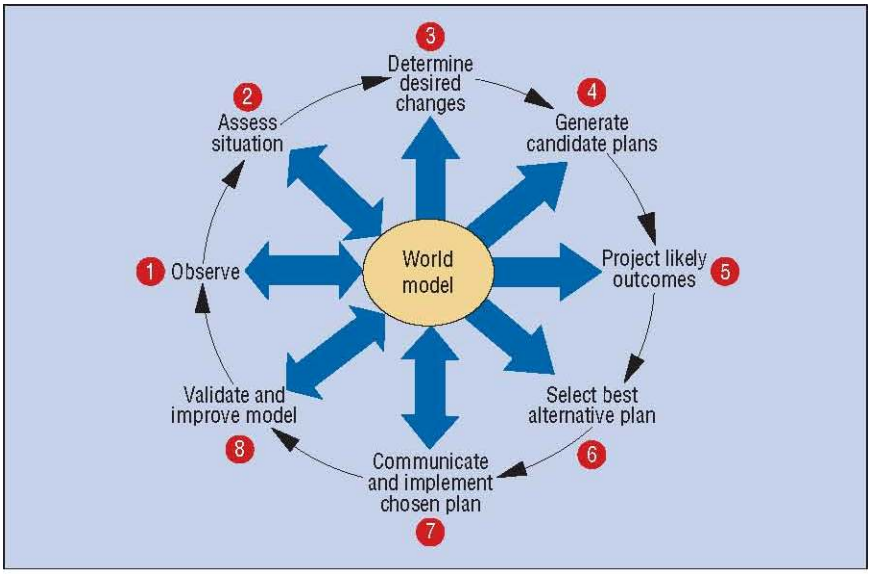

Figure 1. Efficient thought employs eight key functions supported by a world model.

5. Knowledge is an uncertain thing

Aside from definitions, beliefs are never certain

Theorems and "laws" only hold within constrained environments

Theories and "proven methods" only hold within constrained contexts

Every belief can be wrong

Knowledge is built model, upon model, upon model

Knowledge is strengthened when it's vulnerable to disconfirmation and holds up

Knowledge is wrong when it's disconfirmed

6. We can build autonomous machines today, but they have limited learning capabilities

7. We can build learning machines today, but they operate at low levels

Parameter identification, strengthening, tuning

Running experiments over large numbers of candidate operators

Finding and selecting sets of coadapted features in a human-designed feature space (genertic algorithms)

Identifying patterns of co-occurring sets of features or sequences of preidentified elements (as in genome sequencing, linguistic n-grams, grammars for lanaguge)

Decision trees, concept rules, and Bayesian correlations among given sets with given features

8. Few programs have learned new "features" that extend the basic conceptual foundation for machines to reason and learn about

Doug Lenat built a couple of programs that learned new "features" that could describe primitives in mathematics

Hayes-Roth, Klahr, Mostow built a couple of programs that invented "features" to support operationally important distinctions in card games

Image processing systems routinely use low-level feature histograms to partition perceptions into separable categorical clusters (as in different colors, or different textures)

9. Few programs have described operational (behavioral) semantics and domain interactions

Most formal models have been built in non-operational domains (e.g., mathematics, software morphisms)

Most spatial, temporal behavior models have been simple or tailored entirely to a narrow set of behavior objectives (e.g., autonomous driving)

Most automatic programming and program synthesis ignores space and time as well as operational effects (changes to world state)

10, Many planning programs have been written and, given a reasonable semantic model of the world and of the operators, can generate plausible operational sequences

11. Very little work has been done on using expectation failures to generate conjectured changes in belief

The basics are described in Hayes-Roth, "Hyper-Beings" and in earlier Rand papers

Plan justifications would be the main record for analysis

Conjectures would be needed about alternatives

Experiments would be required to ferret out the best plausible fixes

This is the enabler for exponential rates of evolutionary improvement

12. Machines that could evolve and experiment would have special needs and present special challenges

They need autonomy and exercise fields

They ultimately need to engage the rest of the world

They can commit errors and do damage

They need to be monitored, regulated, and constrained

Negative consequences may go unnoticed for some time

The monitors, regulators, and constrainers must also evolve at rapid rates

13. Major R&D efforts should shift now toward the evolutionary adaptive portfolio approaches

Even where engineering processes are manual, the rate of change in off-the-shelf capabilities exceeds the rate of typical system development projects

This is especially true in government and military systems

This is increasingly true in commercial and industrial products as well

The best way to exploit these opportunities is to conjecture products and subject them to test by users

The best way to adapt them is to apply the same ideas as covered in 1-12

Organizations (governments, nations) that cannot capture the opportunities presented by exponentially accelerating components will fall hopelessly behind

Cycle time to market has to be at least as fast as cycle time for component generations (now 6-24 months)

Pragmatics of Doing AI: How would you choose an opportunity and an approach?

1. What type of portfolio are you managing?

How much, how many, how soon?

What kind of success are you seeking?

What kind of failure can you tolerate?

What kind of superiority, over what kind of competition, is needed?

2. Relating the AI technology/methodology to the payoff time-frame

AI approach: Human-engineered v. Evolutionary

Time frames: 2, 5, 10, 20 years

AI APPROACH		PROBABLE TIME TO VALUE			
STRATEGY	**TECHNIQUE**	**2**	**5**	**10**	**20**
HUMAN-ENGINEERED	Rules & facts	✓			
	Expert know-how	✓	✓		
	Sense, plan, act	✓	✓		
	Model, tune, interpret & project	✓	✓		
	Teleoperation & Autonomy		✓	✓	
EVOLUTIONARY	Parametric Design & Optimization	✓	✓		
	Population Trials in Public Arenas		✓	✓	
	Model Improvement Conjectures		✓	✓	
	Model Change Validations		✓	✓	
	Robust Efficient Thought			✓	✓

Workshop Exercises

A. Identify market characteristics where evolutionary AI opportunities exist today

B. Identify technology capabilities and components available now that can implement identified market opportunities

C. Develop a scenario that illustrates 1.5 cycles of evolutionary AI in the chosen market application with the available technology capabilities

Some related research and references

This outline appears at http://faculty.nps.edu/fahayesr/aem.html

Intelligent Systems development without humans in the loop

Hayes-Roth, F. (2006). "Puppetry v. Creationism: Why AI Must Cross the Chasm." *Intelligent Systems* **21(5)** (Sept/Oct): 7-9.

Hayes-Roth, F. (2007). "AI: Ready to leap to neo-Creationism." *IEEE Potentials* **26(5)** (September/October), 20-25

Operationalization (translating advice about how to do things into actual procedures)

D.J. Mostow. A transformational approach to knowledge compilation: replayable derivations of task-specific heuristic search algorithms. In M. Lowry and R. McCartney (editors), *Automating Software Design*, chapter 10, pages 231-259. AAAI Press, 1991.

D. J. Mostow. Design by derivational analogy: issues in the automated replay of design plans. In J. Carbonell (editor), *Machine Learning: Paradigms and Methods*. MIT Press, 1990. Originally published in *Artificial Intelligence 40*: 1-3, September 1989, pages 119-184, Elsevier Science Publishers (North-Holland).

F. Hayes-Roth, D. J. Mostow, and P. Klahr. Knowledge acquisition, knowledge programming, and knowledge refinement. In P. Klahr and D. Waterman (editor), *Expert Systems: Techniques, Tools and Applications*, pages 310-349. Addison-Wesley, 1986.

D.J. Mostow. Why are design derivations hard to replay? In T. Mitchell, J. Carbonell, and R. Michalski (editors), *Machine Learning: A Guide to Current Research*, pages 213-218. Kluwer Academic Publishers, Hingham, MA, 1986. Revised and condensed version of paper in *Proceedings of the 3rd International Machine Learning Workshop*.

D. J. Mostow. Machine transformation of advice into a heuristic search procedure. In J. G. Carbonell, R. S. Michalski, and T. M. Mitchell (editors), *Machine Learning*, pages 367-403. Palo Alto, CA: Tioga, 1983.

F. Hayes-Roth, P. Klahr, and D. J. Mostow. Advice taking and knowledge refinement: an iterative view of skill acquisition. In J. A. Anderson (editor), *Cognitive Skills and their Acquisition*, pages 231-253. Erlbaum, 1981. Presented at 1980 Carnegie Symposium on Cognition, Pittsburgh, PA.

D J Mostow. Machine transformation of advice into a heuristic search procedure (1983). In Machine Learning: An Artificial Intelligence Approach.

Diana Gordon , Devika Subramanian. A Multistrategy Learning Scheme for Assimilating Advice in Embedded Agents. http://www.aic.nrl.navy.mil/papers/1993/AIC-93-016

Genetic Algorithms

J. H. Holland, Adaptation in Natural and Artificial Systems (1975)

http://en.wikipedia.org/wiki/Genetic_algorithm

Evolutionary Algorithms

http://en.wikipedia.org/wiki/Evolutionary_algorithms

Genetic Programming

http://en.wikipedia.org/wiki/Genetic_programming

Interactive Evolutionary Computation

http://en.wikipedia.org/wiki/Interactive_evolutionary_computation

Reinforcement Learning as applied to robots

S. Mahadevan , J. Connell, et al. Automatic Programming of Behavior-based Robots using Reinforcement Learning (1991).
http://web.cps.msu.edu/~mahadeva/papers/aij-obelix

Neural Networks

Biological neural networks

http://en.wikipedia.org/wiki/Biological_neural_network

Synaptic change rule

http://en.wikipedia.org/wiki/BCM_theory

Cortex

Hawkin's memory-prediction in bi-directional hierarchies
http://en.wikipedia.org/wiki/Memory-prediction_framework

Artificial neural networks

http://en.wikipedia.org/wiki/Artificial_neural_network

Hebbian cell assembly theory

http://en.wikipedia.org/wiki/Hebbian_theory

Software & System development

Agile methods

Agile Manifesto
http://agilemanifesto.org/principles.html
Extreme Programming
http://en.wikipedia.org/wiki/Extreme_Programming
Scrum
http://en.wikipedia.org/wiki/Scrum_(development)

Adaptive evolutionary management

http://faculty.nps.edu/fahayesr/aem.html

Iterative Incremental Development (IID)

http://en.wikipedia.org/wiki/Iterative_and_Incremental_Development

Spiral model

http://en.wikipedia.org/wiki/Spiral_model

Balance between agile and "waterfall" approach

http://www.amazon.com/Balancing-Agility-Discipline-Guide-Perplexed/dp/0321186125

Distributed Large-Scale Intelligent Systems (Hyper-beings)

This book also explains how to learn from expectation failures and conjecture new concepts as required

http://www.amazon.com/Hyper-beings-Intelligent-Organizations-Information-Superiority/dp/159113966X

Adaptive Evolutionary Management

Denning, P. J., C. Gunderson, and F. Hayes-Roth (2008). "Evolutionary System Development." *Communications of the ACM* **51**(12): (December): 29-31.

Denning, P. J. and F. Hayes-Roth (2006). "Decision Making in Very Large Networks." *Communications of the ACM* **49** (11)(November): 19-23.

Hayes-Roth, F. (2006). *Event Processing in the Global Information Grid (GIG):Orders of Magnitude Advantage in Information Supply Chains through Context-sensitive Smart Push ("VIRT")*. 2nd Event Processing Symposium, Oracle, Redwood Shores, CA.

Hayes-Roth, F. (2006). *Plenary C2 Policy Panel: Under the avalanche, which way is up?* 2006 Command and Control Research and Technology Symposium, San Diego, CCRP Department of Defense.

Hayes-Roth, F. (2007). Getting ahead of the Avalanche: How everyone can benefit from a near-infinite amount of technology. MESDA's 15th Annual Conference, Maine's Software and Information Technology Industry Association.

Hayes-Roth, F. (2009). Testing Evolutionary Development: Top 10 Good & Bad Ways to Evolve. 2009 DISA Customer Participation Conference, JITC Testing Track, Industry Panel, Anaheim, CA.

Hayes-Roth, F. (2009). "Getting More Bang for the Buck." *IEEE Spectrum* **46**(1): (January) 10.

Hayes-Roth, F., C. Blais, et al. (2008). *How to Implement National Information Sharing Strategy*. AFCEA-GMU C4I Center Symposium: Critical Issues in C4I, George Mason University, Fairfax, VA, AFCEA. (PPT here)