Learning Automata and Need-based Drive Reduction

Ovi Chris Rouly
Department of Computational Social Science
George Mason University
Fairfax, Virginia, USA
orouly@gmu.edu

Abstract—**This paper describes the instantiation of a type of Learning Automata (LA) that is all but unknown. Derived from the later theories of Alan Turing the LA is a dynamic, artifactual model of natural intelligence that uses cybernetic theory to derive machine intelligence. Similarly, the bibliographic references cited here may be unknown to readers unfamiliar with this branch of self-organizing (cybernetic) systems research. Simply put, this research is based on an algorithm described by Turing and named by him the P-Type unorganized machine. The research instantiates the P-Type algorithm in a multi-agent system (MAS) and explores its potential for extensibility. The only other known experiment that used a P-Type machine as a system controller (excluding the work of Turing) instantiated the LA in a single agent system. The current work builds upon that previous experiment; also described here.**

Keywords - Turing machines, P-Type Unorganized Machines, learning automata, multi-agent system, predator/prey simulation.

I. INTRODUCTION

A MAS where foraging behavior, survival tactics, inter-agent socializing, and vehicular hosts steered by a novel, need-based, drive-reducing LA architecture is described. Each "agency" in this MAS was composed of four, independent P-Type unorganized machines acting as individual "agents." At load-time, the user could select the initial number of predatory and prey host icons for instantiation within an enclosed maze. During run-time, the icons were free to move about the maze until they were eaten, their energy levels were depleted, or they adapted, found available food, avoided their predators, and survived. Within a respective agency, one of four LA was selected by afferent conditions to equilibrate over current sensory affect provided by the respective host icon and an isopraxis ethology. All complex behaviors in the hosts were emergent. Appearing as icon hosts with differences in color, size, and markings, predators were distinguishable from prey. The maze was seen from overhead. The program used two physically separate computing platforms: a maze computer and a control computer. The former machine generated the simulation of a multi-baffle, porous-wall (open trellis-like) maze containing the icon hosts. The latter machine generated efferent control signals for the predatory and prey hosts, and received sensory afferents as inputs from the former machine in return. Olfaction, touch, and hunger/satiation were the afferent senses. Motor commands were the efferents.

II. REVIEW

The research reviewed here had four goals: 1) build a simulated, multi-agent colony of predators and consumable prey within an artificial maze, 2) create a method to parameterize and parallelize multiple separate learning automata so that each could independently reduce a specific drive as an individual agent within a more complex combined agency, 3) show that an equilibratory function (operating over sensation and an isopraxis ethology) could enumerate increasingly competent motor and social behaviors, and 4) demonstrate the potential for the system to create simple, extensible models recognizable to social scientists. That work is described in the sections that follow.

A. Goal 1 - Build a Maze.

The first goal was to construct (in software) a multi-agent colony of simulated predators and consumable prey. A simulation that supported 0-5 simulated predators and 0-25 consumable prey was developed. The MAS operated using random, interleaved activation through a control scheme initiated in the control systems computer.

The control scheme can be described as a sensori-motor loop. The iterative process involved: 1) randomly selecting a host (implying a corresponding control agency), 2) determining what agent within that agency would process the environment last sensed by that particular host, 3) retrieving the stored sensations the host encountered when it was last active, 4) processing those sensations, acting/reacting to the maze (and any other entities in the environment) according to or in response to those sensations, 5) receiving stimuli from the maze environment in return, 6) storing those sensed stimuli for use on the next activation cycle, and 7) releasing control to the next host selected by the control computer.

Concern for processor bandwidth and the extensibility of experimental results to mobile robotics applications led to the use of a distributed, two-computer design. Thus, while one computer generated the simulated environment (the maze and the host icons), the other computer supplied the control systems (or agencies) to the hosts, respectively. Ultimately, the colony was placed in a simulated, reconfigurable training enclosure that resembled a maze. A screenshot of the maze immediately after host generation is shown in Fig. 1.

Microsoft Visual BASIC was selected as the compiled language. It offered support for a mixed object oriented and procedural software design architecture, and has well known visual appeal. The software architecture under this version of Visual BASIC supported up to 175 host/agency icon objects. However, only 30 host icons were ever instantiated during this work. Fig. 1 shows 30 of the icons at the start of a simulated run.

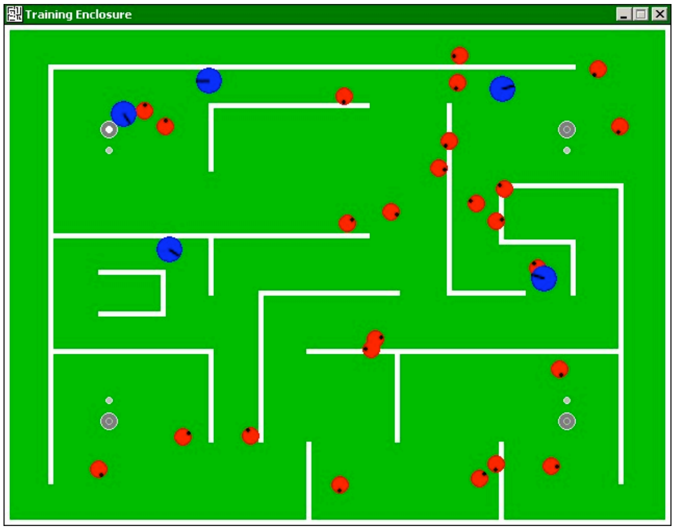

Figure 1. Overhead at the start, the icon hosts were placed randomly. They quickly fell to the floor to begin the simulation. Four gray circles in the floor were prey feed-stations that could emit a simulated odor (only one was active at a time). Predators and prey emitted unique scents of controllable intensity.

In Fig. 1, the hosts are shown placed randomly within the maze. After the simulation began, the icons would quickly fall to the floor of the maze (obviously between the simulated baffles) and would begin to move about. The maximum simulated numbers were one agency per host icon and four agents per agency for a total of 120 agents (120 independent LA) driving the 30 hosts in the maze.

B. Goal 2 – Parameterize the Learning Automata.

The second goal was to create an algorithm that parameterized and parallelized multiple, independent LA and achieved random, interleaved operation of those LA under an agent-based model (ABM) paradigm within a MAS environment. An extensible, fully parameterized model of the Turing P-Type unorganized machine [1] was developed to reduce the individual drives of the agents and satisfy the requirements just named. These P-Type machines are the LA. They correspond directly to the individual agents in the agencies that controlled the hosts. Each agent, in turn, serviced one need-based "drive" that required Hull inspired drive reduction [2].

Since control authority over 30 individual hosts was required, each host had to have its own controller, or control agency. Additionally, it was determined that each agency would be given a set of four independent P-Type unorganized machines (acting as discrete drive reducing agents) and that these lower-level P-Type agents were expected to produce an aggregate control product at the higher-level agency. Finally, as already stated, the activities of each host/agency in the maze needed to be sequenced as part of a random activation scheme over the set of all hosts existing at the time. Fig. 2 shows a block diagram of one control agency. Each LA (or agent) corresponds to one Finite State Machine (FSM)/Transducer block in this diagram. A set of four LA made up one agency.

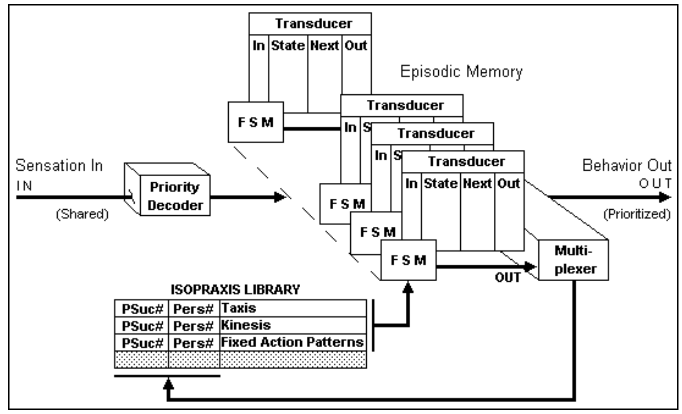

Figure 2. A block diagram of input, output, the four LA/FSM/Transducers, and reconfigurable isopraxis library sources.

C. Goal 3 – Equilibrate Increasingly Competent Behaviors.

The third goal involved showing that an equilibratory [3] function (operating over sensation and an isopraxis ethology [4]) could enumerate competent motor and social behaviors through the P-Types. Clearly, colony members had to equilibrate over sensation and isopraxis ethology in order to reduce simulated physiological, survival-related, and social drives. Simulation of multiple hosts, observation of host behaviors, and inspection of the agency output behavior data files showed that the equilibratory function operated per the requirement and that an enumerated set of emergent behaviors were in fact created and used.

However, a difficulty arose in deriving a quantitative metric for the word "competent." Although not conclusive, one approach to dealing with this problem appeared in an earlier work using similar P-Type LA albeit in a more constrained setting. Rouly [5] described an experiment preliminary to the current work. In that experiment, only one P-Type LA was instantiated per host and only one host was simulated in a maze at a time. That experimental methodology allowed for an analysis of the behavior of the LA under highly controlled conditions. Fig. 3 shows a histogram depicted in that work and is reprinted here.

The figure shows (and describes) the experimental results of a comparative mass trials event involving over 500 host/agent instantiations. The results were statistically significant especially when one considers the data was taken from machines operating as pure "discovery learners" and not stimulus/response, or "interference" learners.[1] [Note: Sidebar

[1] Consistent with Turing (1948, p.45), "interference" learning (or behavior modification), is possible and highly desirable with agencies based on P-Type machines. They respond well to anticipatory stimuli and stimulus reinforcement.

text and line markings have been added to the original histogram to help explain the content.]

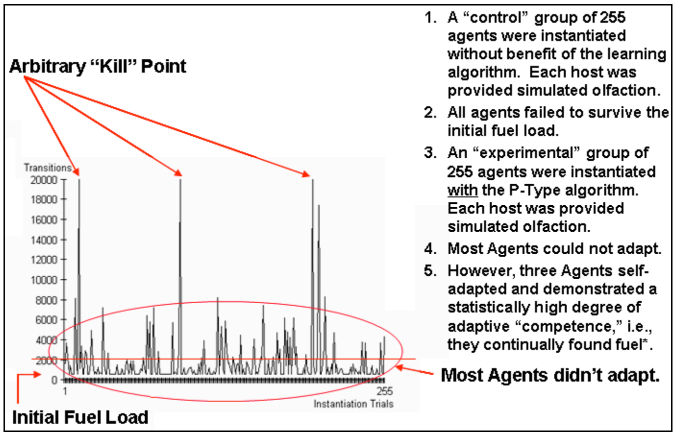

Figure 3. From Rouly 2000, p. 49, Comparison between control and experimental agents.

D. *Goal 4 – An Extensible System Model.*

The fourth goal involved demonstrating the extensibility of the system to model problems recognizable in the social sciences. It is suggested that the system described here could be used to examine questions pertaining to Maslow's *Hierarchy of Needs* [6]. Fig. 4 shows a typical 5-level version of Maslow's pyramid well known by psychologists and educators.

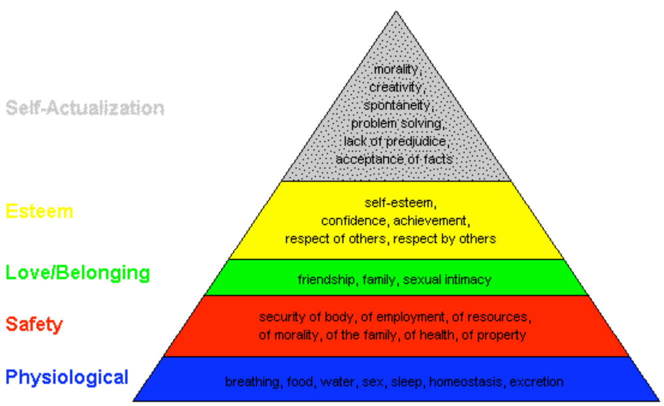

Figure 4. A typical 5-tier version of Maslow's Hierarchy of Needs.

To understand this suggestion, consider the priority decoder in the block diagram of one control agency. (See Fig. 2.) Notice that all sensory stimuli flow through the priority decoder to reach the LAs. In the system developed for this research, the priority decoder dictated which of the four LA/FSM/Transducers would activate "hierarchically." It did this according to programmer hypotheses about the relationship of stimuli to agency/host-needs and "reasonable" reactions. Thus, whether or when an agency/host responded to a stimulus was a function of emergent (learned) agent behaviors and (innate) programmer stimuli-LA mapping. This learned versus "innate" process brought the intuitive appeal of Maslow's pyramid strongly into question more than once during the software development cycle and during experiments.

Fig. 5 shows a typical mapping of stimuli to agency/host heuristics. The mapping was used (for a time) during the development of the current work. The mapping shows what antecedent stimuli triggered the onset of a particular drive, what stimuli tended to reduce a drive (precipitating competent behaviors to emerge), and what stimuli ultimately reduced a drive completely.

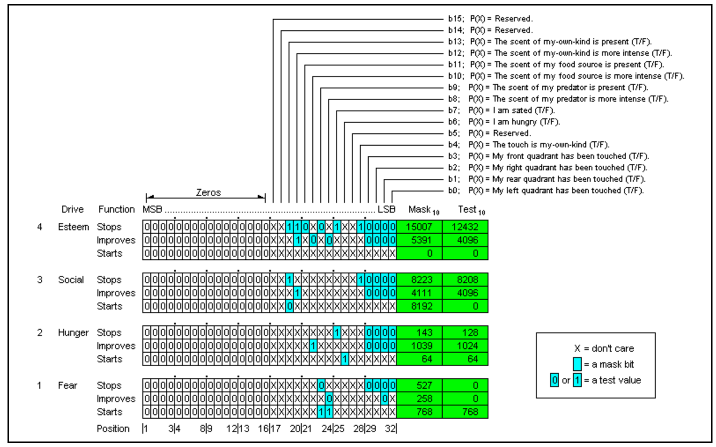

Figure 5. 32-bit afferents mapped to the LA/FSM and the corresponding heuristic implication.

III. Summary

Turing P-Type unorganized machines are adaptive, easy to implement, and are resilient in the face of modification. They provide the machine intelligence researcher a straightforward approach to creating an artificial intelligence, "conditioning" the artificial intelligence/host using behavior reinforcement techniques similar to those used with naturally intelligent systems, and then to go back and examine the response repertoire of the artificial agency/host with regard to new stimuli and stimulus situations. Learning Automata of the type reported here deserve further study and may be extensible to modeling selected problems in the social sciences.

References

[1] Turing, A. M.., Intelligent Machinery: N. P. L. Report of the Controller. (H.M.S.O. 1-20), 1948.

[2] Hull, C. L., Principles of Behavior. New York: Appleton-Century-Crofts, 1943.

[3] Piaget, J., The Equilibration of Cognitive Structures: The Central Problem of Intellectual Development. (Brown, T. & Thampy, K. J., Trans.). Chicago: University of Chicago Press, 1985.

[4] McLean, Paul D., A Triangular Brief of the Evolution of Brain and Law. (from Law, Biology, Culture: The Evolution of Law., Gruter, M. & Bohannan, P., Eds.). Berkeley: The Berkeley Electronic Press, 1982.

[5] Rouly, O. C., Cybernetic Intelligence: A Return to Complex Qualitative Feedback Theory. Las Cruces: New Mexico State University, 2000.

[6] Maslow, Abraham, A Theory of Human Motivation. Psychological Review, 50, pp. 370-396, 1943.