The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

Genetic-Algorithms Produce Individual Robotic-Rat-Pup Behaviors that Match Norway-Rat-Pup Behaviors at Multiple Scales

Chris Sullivan[1], Sanjay S. Joshi[2,4], and Jeffrey C. Schank[3]

[1]Department of Mechanical and Aerospace Engineering, University of California, Davis, USA
[2]Department of Mechanical and Aerospace Engineering, University of California, Davis, USA
[3]Department of Psychology, University of California, Davis, USA

[4]Correspondence: ctljoshi@gmail.com

Abstract: We designed cognitive architectures for individual robotic rat pups using genetic algorithms, with the aim of achieving insight into Norway rat pup behavior. Our genetic algorithms were evolved using only metrics of Norway rap pup behavior (e.g., percent of time spent in corners, along walls, and center of an arena during animal experiments). Robotic rat pups quantitatively matched Norway rat pups at the macro level and additionally qualitatively matched Norway rat pup behavior at the micro behavior scales (corner snooping, punting). The complexity of the resulting deterministic controllers may lend support to previous studies that show random-like control codes (possibly emerging from complex underlying interactions) can produce apparently realistic rat pup behavior below a certain age.

Keywords: genetic algorithm, behavior, biorobotics

1 INTRODUCTION

We combine robotic models of infant Norway rats (robopups), with computer simulation and animal experimentation to study rat pup behavior (Fig. 1). In past work, we showed the remarkable result that simply choosing robopup movement directions at random, regardless of sensory input, produced quite intentional-looking emergent behavior patterns that matched rat pups in both individuals and groups, somewhere in-between 7-10 days of age [1]. Our analysis revealed that body morphology and arena topology interacted with the random control architecture to produce emergent complex behavior. We could not conclude, however, from these results that rat pups in an arena move purely randomly, because we did not investigate the full space of possible deterministic sensorimotor rules. In our current work, we use genetic algorithms to investigate the space of possible sensorimotor rules by artificially designing sensory-dependent deterministic robopup controllers using macro-level topological fitness metrics (e.g., percent of time spent in various regions of an arena during animal experiments). The genetic algorithms allow us to explore a wide range of possible deterministic control solutions. In this paper, we review past work and explain our methods to evolve cognitive codes with genetic algorithms to study a variety of solutions.

2 BACKGROUND

Behavior is influenced by the nervous system, body morphology, physiology, the environment (including the social environment), and interactions among all these elements. Thus, our basic schematic view of behavior is defined in equation (1). Autonomous robots and associated simulations allow systematic variation of the variables of (1) in ways that are often impossible in live animals.

Behavior = *F*(*Internal State, Sensorimotor Rules, Biomechanics, Environment*) (1)

Equation (1) is a schema for developing animal and robotic models and, by including the environment (broadly construed), it is a schema that is likely to result in emergent behavior.

2.1 Rat Pup Experiments

An individual rat-pup experiment consists of placing a single pup in the middle of an arena and videotaping its behavior from above. Figure 1 illustrates a 7-day old rat pup in an arena. Arenas could be manipulated in a variety of ways and the test chamber was configured to study the effects of specific environmental stimuli (e.g., heat, inclines, and light). The video recordings of rats moving in the arena, were then analyzed by extracting digitized video frames at specified time intervals (i.e., 5 secs.) to record the position of the tip of the nose and base of the tail [2].

Fig. 1. 7-day old Norway rat pup during animal observation experiments.

Algorithms were then used to extract metrics from these measurements [3]. Animal experiment results for individual 10-day old pups were reported in Schank et al., 2004 [4]. Rat pups spent time in corners, near walls, and in the center of the arena. In addition, some rat pups visited one corner, some two corners, and others 3 or 4 corners. Sometimes, a rat pup spent an extended period of time in a limited region of the arena. Playback of video from these pup experiments revealed "micro" behaviors, which were limited in time and space. Micro behaviors were easily identified in video playback, but hidden in 2-dimensional trajectory plots. For example, rat pups repeatedly burrowed their nose into corners for variable periods of time (i.e., *corner-snooping*, e.g., see Fig. 1). Rat pups sometimes turned in place in the open or in a corner for periods of time, which is called *punting*. Rat pups also followed walls for periods of time, which we called *wall-following*. These individual rat-pup behavior features also hold for 7-day old pups [1]. We therefore categorized all rat pup behavior as either: (1) macro-level behavior (those relating to overall trajectories which occur on the order of 10 minutes over extended arena space), or (2) micro-level (those that relate to detailed micro-behaviors as described above which occur on the order of seconds to minutes over limited arena space).

2.2 Robotic Rats and Dynamic Simulator

In parallel with the animal experiments, we conducted experiments with robotic rat pups (robopups). The current generation of robopup is shown in Fig. 2. The design of the robopup incorporated biologically inspired and robotic aspects, such as body shape, sensor and actuator location, and computational needs [4]. In brief, the robot's shape was designed to model the basic shape of a rat pup 7-10 days of age. The robot had the same 3:1 length to width ratio as Norway rat pups in the 7 to 10-day age range, and had a similar curved snout (Fig. 1). Rat pups have limited mobility at these ages and primarily use their back legs to push their bodies forward [2]. To model this type of movement, two rear rubber wheels propelled the robot and a supporting passive wheel stabilized the front of the robot.

Before 13 to 14 days of age, rats are blind, deaf, and have limited olfactory capabilities [3]. Thus, infant rat pups rely largely on tactility at these early ages. Robopups incorporated tactility by placing 14 micro-switches at various positions around the robot's perimeter. Most of the sensors were placed near the front of the robopup because a rat pup's primary tactile sensitivity is around the snout area [2]. To detect contact at points where there were no switches, brass strips were connected to the micro switches. Sensorimotor rules were programmed in the robot's microprocessor (Parallax 25 MHz Java stamp 24-pin DIP module). Robots were tested in arenas proportional to the arenas in which rat pups were tested (Fig. 1), and we applied the exact same data extraction and analysis tools for both the robopups and rat pups.

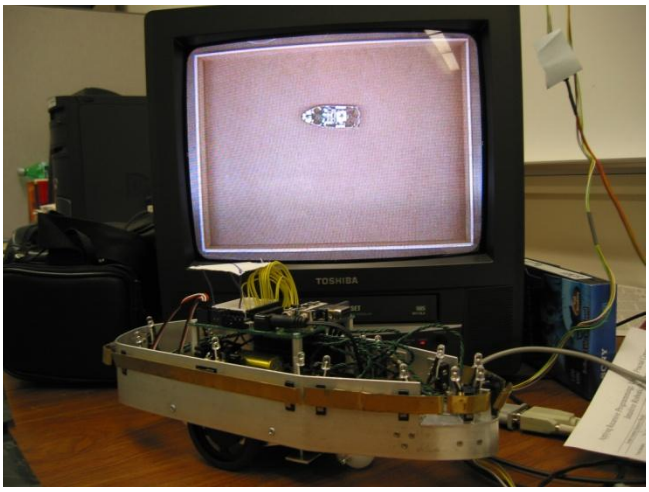

Fig. 2. Robotic model of infant rat pup (foreground), and another robot in testing arena (on monitor).

A MATLAB/Simulink based individual-robot simulation was used in the current study [5]**.** Just as in the actual robots, sensorimotor rules are programmed into the robot simulator. The only differences are the programming language and the computing platform. Model validation studies showed that the simulation produced results very similar to robopups [5].

Several robot controllers were developed for the robopup project to accomplish the goal of modeling rat-pup sensory behavior. Each controller took in information from the robot's tactile sensors and then used that information to send commands to the two robot wheels. The robopup could move forward, back up, or turn (or some combination thereof), depending on the command sent to each wheel.

2.3 Random Control Architectures

In May et al., 2006 [1], we implemented a *random* control architecture as a type of null model to compare to other sensor-driven architectures. In this null model, robots did not use their sensors at all. Instead, every two seconds, robopups *randomly* chose one of ten movements with equal probability: stop moving or one of the seven forward or two back-up directions depicted in Fig. 3.

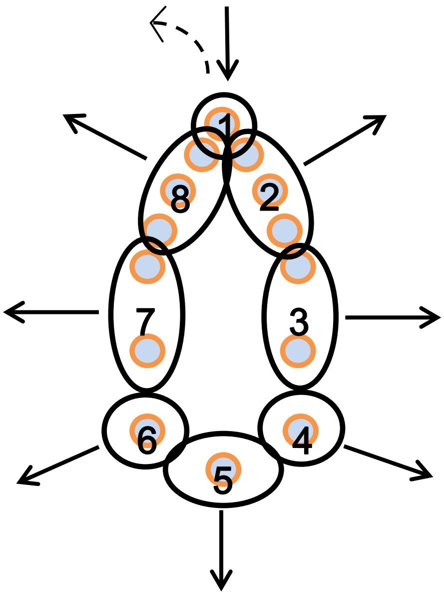

Fig. 3. Illustration of tactile sensor groups on robot, and possible movement directions.

Rat pups can flex their body and turn away from a corner. Rigid-body robopups cannot escape a corner in this way. Therefore, to model the ability of rat pups to escape corners, we implemented *back-up* directions of movement. When moving, the random architecture distributed movements such that a robopup moved forward 78% of the time and backed up 22% of the time. It was surprising and remarkable how well the trajectory plots of "random" robots matched the plots for both 7 and 10 day-old pups. Further analysis showed that the random architecture resulted in good matches somewhere in-between 7-day old and 10-day old rat pups in key macro-level topological metrics, such as the proportion of time spent in corners, near walls, and in the center of the arena [1]. The robopups also exhibited all the micro-level behaviors commonly seen in rat pups including corner-snooping, wall-following, and punting. This surprising result could be explained by the complex interaction of body-shape, arena constraints, and simple cognitive codes all working in unison to allow behavior to emerge.

3 GENETIC ALGORITHM METHODOLOGY

In May et al., 2006 [1], we first designed the controller and then analyzed the ability of the controller to match certain macro-level topological quantitative metrics and micro-level behaviors. In the current work, we reversed the procedure by *imposing* (through artificial selection) the matching of macro-level quantitative topological numerical metrics (percent of time spent in various regions of an arena during animal experiments), and then studied the resulting variety of controllers and their subsequent qualitative micro-level detailed behaviors.

Given that the *random* control architecture, using no sensory information, produced results very similar to the rat pups, a natural question to ask was how well *any* sensory-dependent *deterministic* control architecture could perform in matching pup behavior. To explore the large space of possible deterministic control designs, we employed a genetic algorithm (GA). The evolutionary method used to design controllers depended on three components: (i) macro-level behavior metrics that could be numerically quantified (e.g., percent of time spent in various regions of an arena during animal experiments), (ii) creation of *fitness functions* that evaluated behavioral metrics to score the quality of control designs, and (iii) a method of varying and combining high-scoring control designs to create new control designs (called a *generation* of control designs).

Employing a relatively high-fidelity dynamics model [5] placed computational constraints on our ability to simulate thousands of solutions over hundreds of generations, as is commonly done in GA studies. However, the evolutionary method was still extremely successful at automatically constructing and comparing thousands of computer-designed control codes, many of which closely matched our performance goals and rat pup behavior.

3.1 Control Designs

In this study, the physical design of the robot was held constant; only the motor responses to sensor contact were able to evolve from generation to generation. For each of the eight touch sensor groups on the robot (Fig. 3), a motor response could support up to three consecutive wheel commands (Fig. 4). A wheel command consisted of a left wheel *speed*, a right wheel *speed*, and a time variable to define *duration* of wheel operation. Each sensor's motor response therefore had nine degrees of freedom, bringing the total potential number of variables to 74 (eight sensor groups and an additional two variables to define default left and right wheel speeds when no sensors were currently activated).

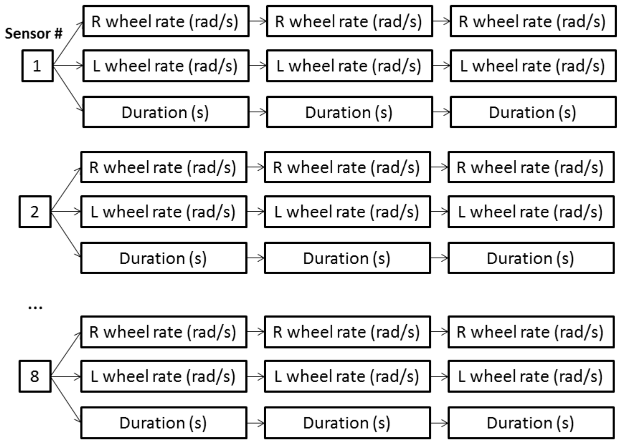

Fig. 4. Variables representing sensori-motor mapping on robotic pup.

It is important to note that a controller design was *not* required to have more than one response for each sensor group. If a controller design consisted of more than one response for a sensor group, the second and third responses were executed sequentially in order, for the amount of time defined in their duration variable (Fig. 4). It was up to the genetic algorithm to select the best combination of the 74 variables that would lead to the most realistic rat pup behavior. The controller genome was then simply a matrix of floating numbers precise to the 2nd decimal place.

3.2 Topological Metric

The underlying fitness metric matched was based on a topological analysis of where infant rat pups spend time in an arena. An average profile of 113 10-day old pups was reported in Schank et al., 2004 by comparing the percentage of total time spent next to walls, in corners, and in the center of the arena during rat pup testing [4]. Every 5 secs during a 10 minute experiment, a pup's position was classified as a *wall event*, *corner event*, or *center event*. A wall event was defined as a snout point next to a wall. A corner event was defined as a snout point within a square area delineated by a corner. A center event was a snout point in the inner arena. We found that rat pups, on average, spent ~60% of time near a corner, ~22% near a wall, and ~18% in the center (Fig. 5). Note that this metric of rat pup behavior does not describe micro behaviors that rat pups exhibited including corner snooping and punting.

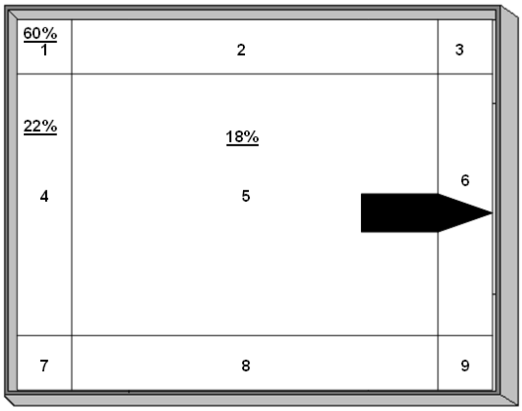

Fig. 5 Topological fitness metrics. Percentages indicate location distributions among all corners (~60%), walls (~22%), and center (~18%) areas averaged over 113 rat pups [4].

3.3 Fitness Function

A linear fitness function was defined as in Eqn. 2, where c is the percent distribution of corner samples (in range from 0-100), c_0 is the desired target percent distribution of corner samples (60% in our case), and the variables w, a, represent the walls and center areas, respectively. The goal then was to maximize this fitness function (c_0 =60, w_0 =22, a_0 =18). Note that the overall fitness function is in the range from 0-1. Even though we refer to the fitness function as linear, the function in (2) is piecewise linear

$$f_{avg} = 1 - \left[\left(\left| \frac{c - c_0}{c_0} \right| \right) + \left(\left| \frac{w - w_0}{w_0} \right| \right) + \left(\left| \frac{a - a_0}{a_0} \right| \right) \right] \Big/ 3 \tag{2}$$

3.4 Genetic Algorithm (GA)

The GA would initially generate 12 controller genomes at random for the first generation of simulated robots. The simulation stored the (x,y) locations of the nose of the robot every 5 seconds to maintain consistency with the original rat pup observations (Section 3.2). For each controller design, three simulation trials were run with different initial headings for a robot initially placed in the center of the arena, since initial orientation often resulted in a substantial difference in robot paths for both deterministic and random architectures. The fitness scores were computed for each simulation and averaged to obtain an average fitness value.

The genetic algorithm employed one of two selection methods to vary control design each generation. Both single parent and crossbreeding techniques were used.

Once the GA evaluated the average fitness of all the simulated robots in a given generation, the top three designs from the population of 12 were selected as a basis for the next generation in the single parent cases. Each of the chosen parent designs created four (slightly varied) versions of itself to produce the 12 children needed for the next generation. In the crossbreeding trials, the process was different. New genomes were created by choosing two parents from the previous generation to essentially blend motor response features of the two parents. Rather than averaging motor responses, the child design would instead pick several motor responses from each parent and the resulting design would be tried. The percentage of cases in which a particular design would be chosen as a parent was proportional to its fitness score.

Mutations could occur at the level of an individual response element ("point" mutations), or an entire sensor response ("bulk" mutations). The mutation rates were 25% for point mutations, and 15% for bulk mutations. Our chosen mutation rates ensured that the children genomes would vary significantly from the parents, but hopefully not so significantly that natural selection was neutralized. Point mutations could modify a single wheel speed by roughly ten percent, or wheel duration by up to 0.1 second. After the new generation of controller designs was mutated, the simulation and fitness function evaluation process started over and repeated for 20 generations.

4 RESULTS

The linear fitness function (2) frequently evolved a control scheme that invoked punting or flailing movement of the robot when a wall was contacted by a sensor. This caused the robot to move across the arena in unpredictable trajectories. In some cases, a no-sensor-contact default curved movement would lead the robot into another wall or corner across the arena, at which point the robot would again spin away from the wall and head out into the arena again (Fig. 6).

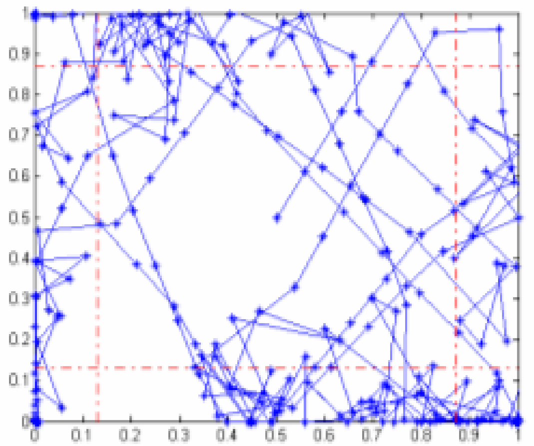

Fig. 6. Representative trajectory of robotic rat pup using a GA solution. This controller achieved an average corner behavior of ~65%, average wall behavior of ~21%, and average center behavior of ~14%.

The best (fully deterministic) linear fitness controller trajectories were quantitatively and visually similar to both infant rat pup trajectories (Schank et al., 2004 [4]) and the previously reported (May et al. 2006 [1]) random robopup trajectories in that the controller produced:

- trajectories that closely matched the macro-level topological corner, wall, and center distributions (by selection with the fitness functions);
- a wide variety of trajectories for a single controller, including trajectories that visited a varying number of corners (depending on a simulated robopup's initial heading);
- typical rat-pup micro-level sub-behaviors like corner snooping, wall-following, and punting
- many trajectories that traversed the center of the arena and crossed the length of the arena to the other side, and whose paths crossed over each other multiple times.

Only the first item above was explicitly designed-for in the GA methodology. The other three items were emergent.

5 CONCLUSIONS

The best GA solutions (in terms of the macro-level topological fitness metric and micro-level rat pup behaviors) incorporated a repeated thrashing at wall contact and a 'random-looking' projection of the robot into the center of the arena, which is interesting for two reasons. First, it supports the idea that a 'random-like' component is needed to match observed behavior, regardless of whether the random-like behavior was created by a truly random controller as in May et al., 2006 [1] or a complex deterministic controller as evolved in this study. As discussed in May et al. 2006 [1], random-like behavior need not result from truly random commands, but in biological organisms, it could be, for example, the result of developing motor systems. Secondly, our results illustrate the potential complexity of sensory-dependent controllers that may be required to produce realistic behavior. The controllers we evolved were complex, with each sensor contact followed by possibly three motor commands in succession. In effect, this work and our past work bounds the rat-pup controller problem at the controller extremes, from a simple random controller (May et al. 2006 [1]) to a complex deterministic controller (current study). Indeed, we have shown that both extremes can produce apparently realistic rat-like behavior for individuals. A promising future line of study would mix deterministic and stochastic controllers. Finally, more investigation needs to be conducted into group behavior. May et al. 2006 [1] showed random-robot behavior metrics intermediate between 7-10 day old pups in individuals and groups. However, a recent study by May et al. 2011 [6], which studied only group behavior, showed metrics that match random-robot group behavior and 7-day old group behavior. But, random-robot group behavior and 10-day old group behavior did not match. This may imply that near 10-days of age, behavior shifts depending on isolation vs. group contexts [3]. However, more investigation with both random and non-random models, and different age rat pups needs to be conducted before any conclusions can be drawn.

ACKNOWLEDGMENTS

This work was supported by the U.S. National Science Foundation under Grant No. 0218927 and by the U.S. National Institutes of Health under Grant 5R01MH065555.

REFERENCES

[1] C. J. May, J. C. Schank, S. S. Joshi, J. Tran, R. J. Taylor, and I. E. Scott, "Rat pups and random robots generate similar self-organized and intentional behavior," *Complexity*, vol. 12, pp. 53-66, 2006.

[2] J. Schank and J. Alberts, "The developmental emergence of coupled activity as cooperative aggregation in rat pups," *Proceedings of the Royal Society of London B*, vol. 267, pp. 2307-2315, 2000.

[3] J. Schank, "The development of locomotor kinematics in neonatal rats: An agent based modeling analysis in group and individual contexts," *Journal of Theoretical Biology*, vol. 254, pp. 826-842, 2008.

[4] J. Schank, C. J. May, J. Tran, and S. S. Joshi, "A biorobotic investigation of Norway rat pups (Rattus norvegicus) in an arena," *Adaptive Behavior*, vol. 12, pp. 161-173, 2004.

[5] R. L. Bish, S. S. Joshi, J. Schank, and J. Wexler, "Mathematical modeling and computer simulation of a robotic rat pup," *Mathematical and Computer Modeling*, vol. 45, pp. 981-1000, 2007.

[6] C. May, J. Schank, and S. Joshi, "Modeling the Influence of Morphology on the Movement Ecology of Groups of Infant Rats (Rattus novegicus)," *Adaptive Behavior*, vol. 19, pp. 280-291, 2011.