# Webworms: Modeling emergent behaviors using LEGO robotics

Matthew D. Riddle, Jeffrey T. Lawson, Matthew C. Jadud
Indiana University
Bloomington, Indiana 47405

## Abstract

The purpose of the Webworms project was to model the synchronous twitching behaviors of a colony of Tiger Moth larvae. Initial video observations of a colony of larvae led to a five phase behavioral model: twitch onset, synchronization, continuation, cessation, and a resting phase. A working model of the emergent behaviors exhibited by the caterpillar colony was hypothesized and implemented in hardware and software, making use of the LEGO® Mindstorms™ Robotic Invention System™ as a rapid-prototyping tool.

## 1 Purpose

The purpose of the Webworms project was to model the synchronous twitching behaviors of a colony of Tiger Moth larvae using the LEGO® Mindstorms™ Robotic Invention System™. This twitching, which appears to be a defensive mechanism, begins when a fly lands on their web. It begins with a violent twitching that propagates through the colony like a wave. After several seconds of twitching the colony stops, then after a brief delay they start again and repeat the cycle. As the cycle continues the twitching phase decreases in length, while the delay or resting phase grows longer, as illustrated in figure 1.

The LEGO® Mindstorms™ Robotic Invention System™ comes with a programmable "brick" which is programmed on a PC, and completed programs are downloaded to the "brick" using infrared communications. In addition to an assortment of LEGO® pieces, two touch sensors, two low torque motors, and one light sensor are included in the kit for interacting with the world. We chose this kit because of its value as a rapid-prototyping platform for developing simple and moderately complex robots.

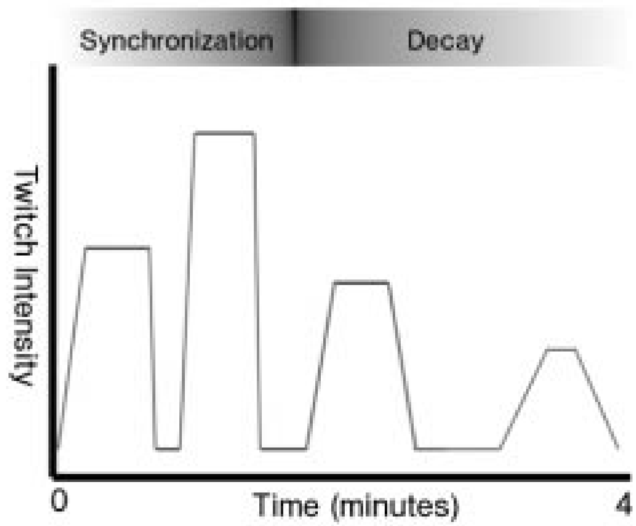


Figure 1: A graphical representation of the 5-phase model for Tiger Moth colony twitching behaviors.

## 2 Method

This research in building a working model for the behaviors of the Tiger Moth larvae on their web was an evolutionary, learning process. It began, and continues to be, a learning opportunity for those involved, both as participants and as mentors.

### 2.1 Brainstorming

Upon beginning our work, we were presented with a video of a Tiger Moth larvae colony by Professor Jonathan Mills (figure 2). We studied the video, and with Professor Mills, hypothesized a five-phase model representing the caterpillar's actions on the web. These five phases make a repeating cycle, beginning with a twitch onset, apparently triggered by a fly landing on the colony's web. This twitching begins to propagate across the web, after which they synchronize, the second phase. The synchronized twitch

Figure 2: The Tiger Moth colony

duration decreases as the caterpillars continue through the cycle. The caterpillars then stop twitching (again, in near synchrony), and appear to "rest;" we have named this the "resting phase". The cycle repeats for several minutes, each phase growing shorter as the worms continue. Interestingly enough, the worms do not stop their progression through the cycle as soon as the fly leaves; subsequent cycles are speculated to be a precautionary measure.

With this information, we brainstormed a number of approaches that we might take in constructing the webworm colony in LEGO®. The question was "how do we want to model the communication between caterpillars?" We considered many approaches, including some which involved physically connecting the robots with elastic material; we eventually settled on making use of the Mindstorm's built-in IR transmitter and receiver to model the communication between caterpillars on the colony's web.

## 2.2 Exploration

One of the first tasks Mr. Riddle worked on was the question of how to make the robots "twitch." From previous experience with ROBOLAB™, how to write a program that produced this behavior was not a challenge, but learning C was. Dave Baum's NQC[Bau99] was chosen for its advantages over ROBOLAB™. While the visual language is great for testing ideas, it is difficult to implement complex ideas in a graphical language. For this reason, we adopted NQC for our experiments in developing a colony of robotic caterpillars.

Mr. Riddle's first attempts at looping in C focused on making the robot do simple squares and spirals. Spirals were chosen for the need to change what the robot is doing every time it turns or moves forward. The goal in this early stage of exploration was to keep the programs as simple as possible. At the same time, he also explored timing and issues involved with rotation without rotation sensors. The first attempts at "twitching" were done by timing the movement of the robot such that when finished, it would end up facing the same direction as it began. This proved to be difficult, as compensations were needed for the momentum of the robot and its rapid changes in direction.

Mr. Lawson was, at the same time, exploring how to enable communication between the robots. The problem we faced was finding a way for four robots to communicate with each other, when they all required line of sight. We tried several different robot configurations, none of which solved the problem. This required us to take a step back and think about the problem. Since the IR signal is simply light, we decided that it could be reflected off some type of material. After some experimentation we decided that the best material for this would be something mirror-like, so we constructed a wall surrounding the robots and covered it with aluminum foil. The foil, even though not perfect, did bounce the signal well enough for all robots to communicate. Also, since the pen was fairly large, it allowed for freedom of movement and space for the robots to interact.

Once issues of programming and position had been dealt with, we implemented our first version of the 5-phase model. This was a hierarchical "master/slave" system, where only one member of the colony gave instructions, and the rest followed them. This had many problems, though. When compared to the video, our colony did not quite "look" right. We initially thought that there might be "lookout" caterpillars in the colony who announced the presence of a fly or other predator. After many viewings, careful examination, and much discussion, we began thinking about the colony in terms of the individual worm. At this point, Mr. Jadud recommended we read *Turtles, Termites, and Traffic Jams*[Res94], where we connected our ideas with the notions of emergence and autonomous agents. By moving to this way of thinking about the robots, we achieved a great deal of interaction between them. They were no longer mindless machines doing exactly what they were told to do, but instead interacted and communicated with each other much like a real biological system. This brought the robots one step closer to exhibiting emergent behaviors.

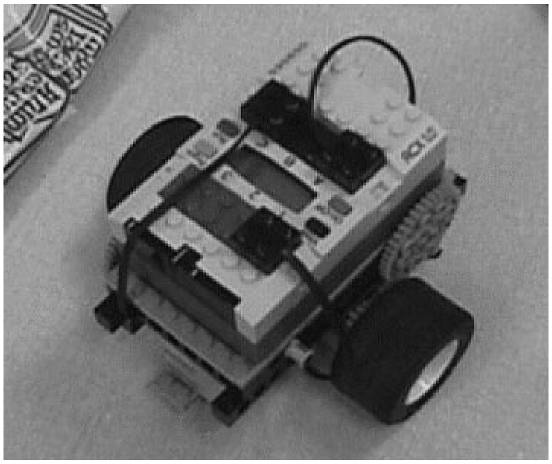

Figure 3: The robotic caterpillar, top view.

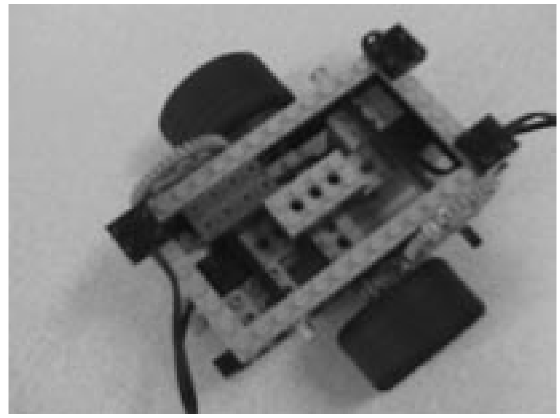

Figure 4: Robotic caterpillar, without the RCX.

### 2.3 Evolution

Examination of the video and our prototypes led us to make another hypothesis. We observed that each of the larvae were performing one simple task, a twitch. When taken as a whole, the colony exhibited a synchronization of movement, which was very different than the individual worm taken alone. We hypothesized that this twitching cycle was an emergent behavior where each worm could be viewed as an individual agent in the system. That is to say that each of the larvae acted on their own and followed a few simple rules which, when performed in large numbers formed a new, more complex behavior. This is similar to the behavior exhibited by a school of fish or a flock of birds. Each of the animals appears to follow a few simple rules: stay close to your neighbor, but don't run into your neighbor. When one animal turns, the others follow, so they all seem to know what the others are going to do, when they are merely following the rules.

This led us to rethink our program design. In the previous prototype, movement of the robots was controlled through hard-coded timings. In doing this, the program controlled the behavior that was observed, and did not result in a very natural looking "twitch." Rotation sensors were added to the robot webworms to allow for finer control of the robots' motion. With the rotation sensors there was the ability to measure how far the robot moved and was able to travel back to its starting point; this allowed for a quick, thrashing, twitch like the larvae's to be implemented.

Our latest prototype is based on the theory of autonomous agents, and we have constructed our third colony of LEGO® robots. Each of the robots is completely independent of one another, yet they are controlled by the same program, making each of them identical. They all have the ability to sense a fly (an IR event), and all have the ability to broadcast to the other robots news of such an event. Therefore, no one robot is the "master," like in our previous approaches. Our goal in this approach was to make the program as simple as possible, while keeping our hypothetical model in mind. We strove for making each robot independent of the others, and therefore allowing for the emergence of behavior from the group, and not our programming.

## 3 Results

One of our most significant results was the construction of two working models of the webworm colony. These models evolved through four robots and six different programs for their control over the course of one semester (figures 3,4,5).

Along with the tangible creations, we have three significant learning outcomes from our experiments in colonial robotics. The first stage of the research was a learning phase, where we debated what tools to use, what approaches might work, and how to use these new tools. Mr. Riddle and Mr. Lawson both worked at transferring their knowledge of programming from their experience with ROBOLAB™ to Dave Baum's Not Quite C. Most of the group's frustration came not from syntactic issues, but instead questions like "what was the command to make the motors stop?"

The second stage of our work was where Mr. Lawson began reevaluating our ideas about how the colony

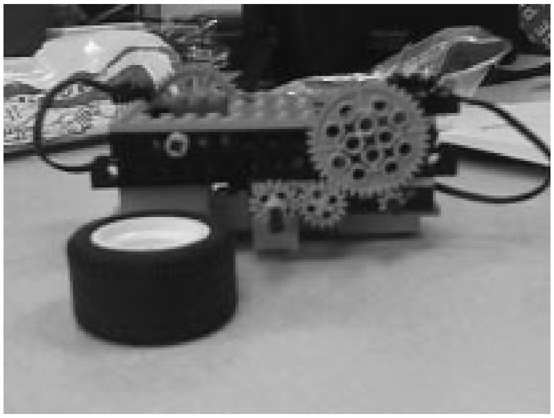

Figure 5: Robotic caterpillar, side view.

might actually function. His insight led to Mr. Jadud's suggestion that a reading of *Turtles, Termites, and Traffic Jams* might be appropriate, and further focus their thoughts. This change in thinking from a global world-view to a perception of the system as a collection of individuals is a very powerful one to make.

Lastly, we have a working model for how undergraduates and graduates can work together at a large, liberal arts university. With minimal budget, an inviting, challenging, and exploratory research environment has been created where majors and non-majors alike can explore advanced concepts in computer science and robotics.

# 4 New or breakthrough aspect of work

This research is driven by the interest and enthusiasm of undergraduates and graduate students working together at Indiana University Bloomington. As a liberal arts institution enrolling over 30,000 students per year, it is difficult to imagine having an inexpensive research setting where each student has access to multiple robots for exploring issues in synchronization, communication, and emergent behaviors in systems. It is exactly this environment that we have managed to create.

Obtaining easy access to powerful computers and the Internet is no longer a difficult feat; likewise, LEGO®, in conjunction with the MIT Media Labs, has provided a rapid-prototyping environment for exploring powerful ideas in robotics in settings where it would have been considered impossible a few scant years ago.

# 5 Conclusions

The LEGO® Mindstorms™ Robotics Invention System™, although marketed as a children's toy, is an accessible, approachable, and cost-effective platform for university-level research in robotics. It is our intent to continue this research, hopefully expanding to more robots in the colony, and continue to bring new, interested students into the project as opportunities to do so present themselves.

http://www.indiana.edu/~legobots/index.html


## Acknowledgments

Many thanks to Professor Jonathan Mills for his continued support of this work and the students of the Adaptive Systems Laboratory in the Department of Computer Science at Indiana University Bloomington. Thanks are also due to Greg Hanek for his input throughout the project, and Jeff Schank for providing the original video documentation of the Tiger Moth's odd, twitching behavior.


# References


[Bau99] David Baum. Nqc programmer's guide v2.1. http://www.enteract.com/~dbaum/nqc/doc/NQC_Guide.pdf, 10 1999.

[Res94] Mitchel Resnick. *Turtles, Termites, and Traffic Jams*. MIT Press, 1994.