MASSACHUSETTS INSTITUTE OF TECHNOLOGY
MEDIA LABORATORY

EPISTEMOLOGY AND LEARNING GROUP

E&L Memo No. 8
September 1990

LEGO/Logo:
Learning Through and About Design

Mitchel Resnick
Stephen Ocko

Abstract

Most classroom problem-solving activities focus on *analytic* thinking: decomposing problems into subproblems. Students rarely get the opportunity to *design* and *invent* things. In this paper, we describe how LEGO/Logo, a computer-based robotics environment, supports a variety of design activities. We examine how students using LEGO/Logo can learn important mathematical and scientific ideas *through* their design activities, while also learning *about* the design process itself.

LEGO/Logo:
Learning Through and About Design

Mitchel Resnick and Stephen Ocko
MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139
mres@media-lab.media.mit.edu
ocko@media-lab.media.mit.edu

Abstract

Most classroom problem-solving activities focus on *analytic* thinking: decomposing problems into subproblems. Students rarely get the opportunity to *design* and *invent* things. In this paper, we describe how LEGO/Logo, a computer-based robotics environment, supports a variety of design activities. We examine how students using LEGO/Logo can learn important mathematical and scientific ideas *through* their design activities, while also learning *about* the design process itself.

Introduction

Problem solving can be divided, very roughly, into two categories: analysis and design. Analysis involves decomposing problems into simpler subproblems, typically with the help of formalized rules. Design is different in several ways. In design, the problem goals are typically ill-structured; defining the problem is part of the designer's job. Moreover, there is a somewhat fuzzy sense of what it means to "solve" a design task. Rather than seeking *optimal* solutions, designers typically seek *satisficing* solutions [Simon 1969]—that is, solutions that roughly satisfy a given set of constraints.

Design is important in almost all fields of human activity. Of course, an architect uses design skills when preparing a blueprint. But so does a writer when writing a report, and a manager when restructuring an organization. Given the central role of design in human activity, one would expect design to play an important role in school classrooms. But it doesn't. In the minds of many educators, the ill-structured nature of design activities makes them ill-suited for the classroom. Design activities, they complain, are difficult to "manage" and to evaluate. As a result, students rarely get the opportunity to design, to build, to create, to invent.

This paper explores the role of design in the classroom. In particular, it focuses on the use of LEGO/Logo, a computer-based robotics system that supports a variety of design activities. The paper examines how students using LEGO/Logo can learn

important mathematical and scientific ideas *through* their design activities, while also learning *about* the design process itself.

LEGO/Logo

LEGO/Logo links the world of LEGO construction with the world of the Logo programming. In using LEGO/Logo, children start by building machines out of LEGO pieces, using not only the traditional LEGO building bricks but newer pieces like gears, motors, and sensors. Then they connect their machines to a computer and write computer programs (using a modified version of the programming language Logo) to control the machines. For example, a child might build a LEGO merry-go-round, then write a Logo program that makes the merry-go-round turn three revolutions whenever a particular touch sensor is pressed.

LEGO/Logo builds on several decades of research on computers and children. In the late 1960's, Seymour Papert and colleagues at MIT developed Logo as a programming language for children [Papert 1980]. The most popular application of Logo involved the "floor turtle," a simple mechanical robot connected to the computer by a long "umbilical cord." Logo included special commands like `forward`, `back`, `left`, and `right` to control the floor turtle. For example, children would type `forward 50` to make the turtle move forward by 50 "turtle steps," or `right 90` to make the turtle turn right through 90 degrees.

With the advent of personal computers, the Logo community shifted its focus to "screen turtles." With personal computers, children still use commands like `forward` and `right`, but these commands control small graphic images on the computer screen, not actual mechanical robots. Screen turtles are much faster and more accurate than floor turtles, and thus allow children to create more complex graphic effects.

LEGO/Logo brings the turtle back off the screen, but with several important differences from the early days of the floor turtle. First of all, LEGO/Logo users are not given ready-made mechanical objects; they build their own machines before programming them. Second, children are not restricted to turtles. Children have used LEGO/Logo to build a wide assortment of creative machines. A few examples: a programmable pop-up toaster; a "chocolate-carob factory" (inspired by the Willy Wonka children's stories); a machine that sorts LEGO bricks according to their lengths; and an "ejection bed" that automatically tosses its occupant onto the floor when the sun shines through the window in the morning. Working on projects like these, children experiment with many different types of design:

structural design, mechanical design, software design. LEGO/Logo might be viewed as a "multi-media construction kit," allowing students to build and create in several different (though interconnected) media.

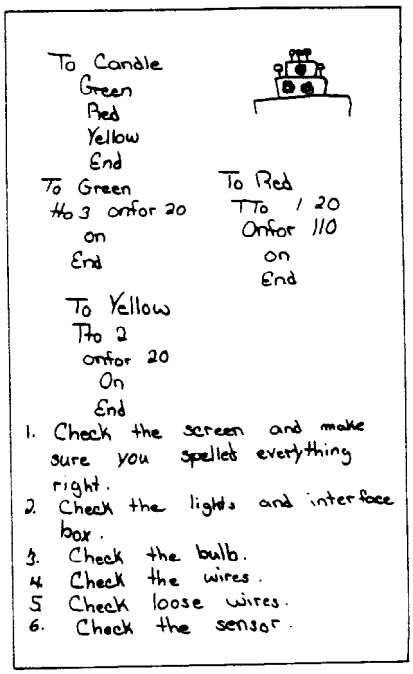
The LEGO/Logo system includes new types of LEGO blocks for building machines, and new types of "Logo blocks" for building programs. On the LEGO side, there is an assortment of gears, pulleys, wheels, motors, lights, and sensors. For example, there are optosensors that report when they detect changes in the level of light, and touch sensors that report when they are pressed. The computer communicates with LEGO devices through a custom-designed interface box, which connects to a slot card in the computer. Information flows through the interface box in both directions: children can send commands to LEGO motors and lights, and receive status information from LEGO sensors.

As its programming language, LEGO/Logo uses an expanded version of Logo. Students can use any of the traditional Logo commands and control structures (such as `forward`, `right`, `if`, and `repeat`), plus any of 20 new commands added specially for the LEGO environment. The new commands include words like `on` and `off` for controlling LEGO motors and lights, and words like `sensor?` for getting information from LEGO sensors.

Just as students can build increasingly complex machines by snapping together LEGO bricks, they can build increasingly complex computer programs by "snapping together" Logo commands. Imagine, for example, a LEGO car with a touch sensor on the front. A student can write a program called `car` that turns the car on, waits until the car bumps into something, then turns it off. The program would look like this:

```
to car
on
waituntil [sensor?]
off
end
```

When we work with students, we attempt to create an "Inventor's Workshop" environment. We show students copies of actual patent drawings by famous (and some not-so-famous) inventors, and we encourage them to keep "Inventor's Notebooks" to document their own designs. In some classes, we have established a system of "LEGO/Logo patents," awarded to students who document their "inventions" with drawings and descriptions.



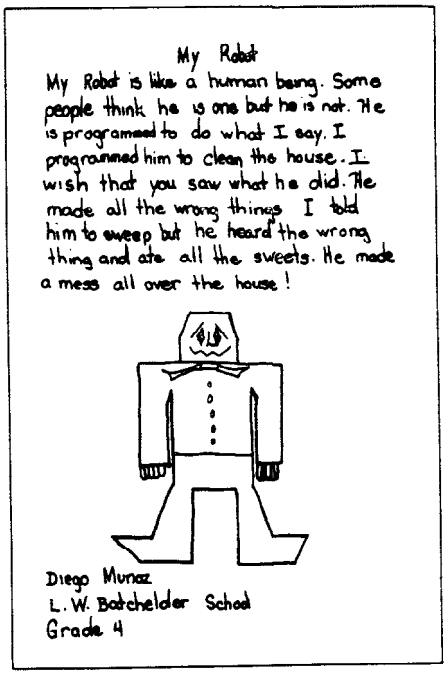
To Candle
Green
Red
Yellow
End

To Green
Mo 3 onfor 20
on
End

To Red
TTo 1 20
Onfor 110
on
End

To Yellow
Tto 2
onfor 20
On
End

1. Check the screen and make sure you spelled everything right.
2. Check the lights and interface box.
3. Check the bulb.
4. Check the wires.
5. Check loose wires.
6. Check the sensor.



My Robot

My Robot is like a human being. Some people think he is one but he is not. He is programmed to do what I say. I programmed him to clean the house. I wish that you saw what he did. He made all the wrong things I told him to sweep but he heard the wrong thing and ate all the sweets. He made a mess all over the house!

Diego Munoz
L.W. Batchelder School
Grade 4

Students use their Inventor's Notebooks in many different ways (see above). Some students make preliminary sketches of their machines. Others make careful mechanical drawings of their constructions and write elaborate instructions on how to use the machines. Still other students use their notebooks to write stories about their machines. Indeed, we find that LEGO/Logo is a rich environment not only for math, science, and design, but also for language arts, since students are often interested in writing about the machines that they have built.

Learning Through Design

LEGO/Logo activities can be organized in many different ways. For example, a teacher could tell each student in the class to build a LEGO car, according to particular "building instructions." Then, the teacher could tell the students to perform a prescribed set of experiments on the car. Certainly, the students would learn something. But such a "recipe-based" approach runs counter to the ideas that informed the development of LEGO/Logo.

In our experience, design activities have the greatest educationally value when students are given the freedom to create things that are meaningful to themselves (or others around them). In such situations, students approach their work with a sense of caring and interest that is missing in most school activities. As a result, students are more likely to explore,

and to make deep "connections" with, the mathematical and scientific concepts that underlie the activities. This idea is at the core of Papert's theory of *constructionism* [Papert 1986].

Our work with LEGO/Logo supports this idea. LEGO/Logo projects typically involve a variety of mathematical or scientific concepts—such as fractions, friction, and mechanical advantage. In many cases, the students have previously "learned" these concepts in the classroom. But students seem to gain a deeper understanding when these concepts are embedded in meaningful design activities.

Particularly striking are the experiences of students who have been unsuccessful in traditional math and science activities. Quite a few of these students thrive in the LEGO/Logo environment. These students often have strong design and mechanical skills, but many have been frustrated by the analysis-centered approach of traditional math and science classes. As a result, their teachers typically view them (and they typically view themselves) as poor learners. In some cases, LEGO/Logo experiences have had a profound effect on how these students view themselves as learners (and on their subsequent performance in other classroom activities).

Example 1: Mathematics through design. George, a third-grade student, began his LEGO/Logo work by building a simple LEGO car. First, he connnected the car's motor to a battery box and watched the car roll forward. Next, he connected the motor to the computer and began experimenting with some of the new Logo commands. We explained the commands `on`, `off`, `onfor` (which takes an input and turns on the motor for a designated amount of time), and `rd` (for reverse direction). After a while, George put several commands together in the following expression: `repeat 4 [onfor 20 rd]`

When George executed this expression, the computer turned on the motor for two seconds (`onfor 20`), reversed the direction of the motor (`rd`), then repeated those commands three more times. The result: the car moved forward and back, then again forward and back, completing two forward-back cycles.

Next, George changed the numerical input to `repeat`. He tried `repeat 3` and `repeat 6` and `repeat 7`. After this experimentation, George noticed a pattern:

> *When I use an even number, the car ends up where it began. When I use an odd number, it ends up away [from where it started].*

George paused for a moment and then added:

> *So that's why there are even and odd numbers!*

Clearly, George had previously learned about even and odd numbers in the classroom. But George's experimentation with the LEGO car provided him with a new (and more personally relevant) representation of the concept. Moreover, the LEGO activity allowed George to *relate* to numbers in a new way: he *played* with the ideas of even and odd. This new relationship with even and odd numbers helped George develop a new level of understanding.

Example 2: Science through design. In some classes, we suggest that students begin their LEGO/Logo work by building motorless "Soap Box Derby" cars. We set up a ramp in the classroom, and students race their cars down the ramp. Some students see a challenge: they want to make their cars go as fast and as far as possible. They measure how far their cars go beyond the bottom of the ramp, then redesign their cars to try to make them go farther.

As students discuss why some cars go further than others, we encourage students to look for places where "pieces are rubbing together" (for example, gears rubbing against other gears, or axles rubbing against the girders that support them). Before long, discussions about *friction* become part of the standard discourse between us and the students—and, more importantly, among the students themselves. In one class, a student removed some pieces from his car, explaining (correctly) that "more weight makes things tighter, you know, more friction." Other students give similar explanations. Even after students finish working on their cars and start designing other LEGO/Logo projects, many of them continue to talk about friction in describing and analyzing the motions of their new machines.

As with even and odd numbers, friction is not a new concept for these students. But the LEGO/Logo activities provide a more meaningful context for students to talk about and think about friction. Students *want* to make their cars go farther, so friction assumes a new relevance. In short, students *care* about friction. At first, students' models of friction are rather fragile, and their explanations are often misleading. But as they work on their LEGO/Logo projects (and participate in a "culture" where friction is a standard part of the discourse), students gradually construct richer models of how friction influences the motions of their machines.

diSessa [1986] describes science learning as a *re-experiencing* process. Children do not learn a new concept when they are taught the definition. Rather, they must experience and re-experience the concept in different contexts. Through these experiences, children gradually reorganize their intuitions into more complete models. In this case, students, by

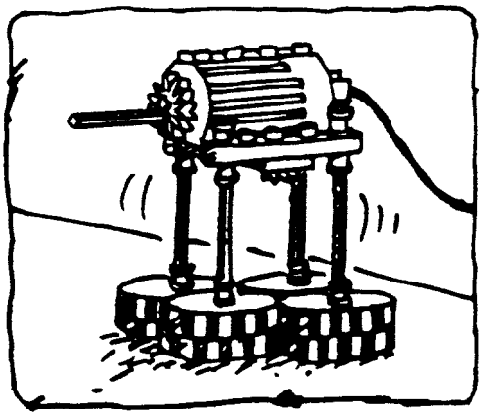
exploring the motions of LEGO/Logo machines, seem to develop richer and more robust models of friction.

Learning About Design

While working on LEGO/Logo design projects, students can learn not only about mathematical and scientific ideas, but about the process of design itself. In particular, students can learn important *heuristics*, or rules-or-thumb, for dealing with design tasks. Such heuristics are particularly important when working in complex design environments like LEGO/Logo, in which the designer needs to consider several different interrelated design tasks.

As an example, consider the "vibrating walker" developed by Nicky, a fourth-grade student at a Boston public elementary school. Nicky started by building a car out of LEGO. After racing the car down a ramp several times, Nicky added a motor to the car and connected it to the computer. When he turned on the motor, the car moved forward a bit—but then the motor fell off the body of the car and began vibrating across the table.

Rather than trying to fix this bug (or giving up since his car had "failed"), Nicky became intrigued with the vibration of the motor. He began to wonder whether he might be able to use the vibrations to power a vehicle. In effect, he decided to turn the vibrations from a bug into a feature.



Nicky mounted the motor on a platform atop four "legs" (LEGO axles). After some experimentation, Nicky realized that he needed some way to amplify the motor vibrations. To do that, he drew upon some personal experiences. Nicky enjoyed riding a skateboard, and he remembered that swinging his arms gave him an extra "push" on the skateboard. He figured that a swinging arm might accentuate the vibrations of the motor as well. So Nicky connected two LEGO axles with a hinged joint to create an "arm." Then, he placed a gear

on the motor and inserted the arm slightly off-center in the gear. As the gear turned, the arm whipped around—and amplified the motor vibrations, just as Nicky had hoped.

In fact, the system vibrated so strongly that it frequently tipped over. A classmate suggested that Nicky create a more stable base by placing a LEGO tire horizontally at the bottom of each of the legs. Nicky made the revision, and his "vibrating walker" worked perfectly. In fact, Nicky was even able to steer the walker. When the motor turned in one direction, the walker vibrated forward and to the right. When the motor turned in the other direction, the walker vibrated forward and to the left.

Next, Nicky set out to make the walker follow a black line on the table top. He attached a LEGO light sensor (pointing down) at the front of the walker. When the walker passed over a black line, the sensor reported `true`. With a bit of assistance from us, Nicky wrote the following program to make the walker follow the line. (We have "cleaned up" Nicky's code to make the program more readable.)

```
to follow
look-for-line
go-past-line
reverse-direction
follow
end

to look-for-line
waituntil [floor-color = "black]
end

to go-past-line
waituntil [floor-color = "white]
end

to floor-color
if sensor? [output "black]
if not sensor? [output "white]
end
```

When the `follow` procedure is executed, the walker veers in one direction until it "finds" the line, continues in that direction until it passes over the line, then reverses the direction of its motor and repeats the process. As a result, the walker weaves back and forth over the line, making a bit of forward progress with each cycle.

What did Nicky learn through this project? For one thing, he gained an introductory understanding of some specific engineering concepts. In building the walker, Nicky ended up with an appreciation for both the constructive uses and the destructive potential of

vibration in mechanical systems. And in programming the walker to follow the line, Nicky explored basic ideas of feedback and control. Nicky used these same ideas in a later project, when he programmed a LEGO "turtle" to find its way out of a large cardboard box.

Equally important, Nicky gained a sense of the *process* of design. In building the walker, Nicky used an impressive array of *design heuristics*. Among Nicky's heuristics:

- *Take advantage of the unexpected.* When the motor fell off of his car, Nicky did not see it as a sign of failure. He saw it as an opportunity. He was on the lookout for unexpected events, and took advantage of them when they happened.

- *Use personal experience as a guide.* When Nicky needed to amplify the vibrations of the motor, he relied on knowledge of his own experiences and body movements.

- *Try using materials in new ways.* The designers of LEGO bricks probably did not envision LEGO axles used as arms or legs. Nor did they imagine that LEGO wheels would be turned 90 degrees and used as feet. But Nicky did not feel constrained by standard usage.

- *Collaborate with others.* When the vibrations kept tipping the walker over, Nicky was uncertain how to solve the problem. So he consulted with a classmate who had a reputation for mechanical-design skills. The collaboration was a success. Such collaborative efforts are particularly important in multidisciplinary activities like LEGO/Logo.

In discussing LEGO/Logo projects, we encourage students to think and talk explicitly about such design heuristics. In many cases, we believe that LEGO/Logo activities have helped students develop a more principled approach to design and invention.

Beyond Hands-On

What makes LEGO/Logo successful as a design environment (and as a learning environment)? Certainly, the materials (LEGO bricks and Logo software) are part of the answer. But only part of the answer. Good tools and materials are necessary but not sufficient for effective design (and effective learning). Our experiences with LEGO/Logo have highlighted several other key ingredients for creating rich design environments for children:

- *Put children in control*. In many hands-on activities in the classroom, students re-create someone else's experiment. Our LEGO/Logo classes have been most successful when children formulate their own designs and experiments, and work on projects that they care about personally.

- *Offer multiple paths to learning.* Not everyone wants to start by building a car. It is important to allow students to approach design projects from different directions

and different perspectives. With LEGO/Logo, some students start with mechanical design, others with programming, still others with architectural aesthetics. Some build moving vehicles, others build moving sculptures. But providing multiple paths does not mean that each student gets "stuck" in his or her own niche. In fact, we have found that many students use their initial "regions of comfort" as a foundation from which to explore other areas—areas that might have seemed intimidating in isolation.

• *Encourage a sense of community.* In LEGO/Logo workshops, we encourage groups to share ideas, designs, and actual constructions—and to critique one another's designs. In this way, students combine "hands-on" activities with what Seymour Papert has termed "heads-in" activities. Moreover, students get a deeper sense of the way in which real designers go about their work, as part of a community of designers.

Implementing these strategies is not easy. There are many unanswered questions—such as how to help students "break away" from their initial "regions of comfort." And coordinating open-ended design activities is a challenge for any teacher. Indeed, organizing an "Inventor's Workshop" is far more difficult than delivering a lecture on mechanical advantage, or developing a step-by-step hands-on lesson. As Dewey [1938] noted more than a half-century ago (in words that still ring true today): "The road of the new education is not an easier one to follow than the old road but a more strenuous and difficult one." But it is a road well worth taking.

Acknowledgments

Many of the ideas in this paper are based on collaboration with Seymour Papert. We are grateful to David Cavallo for sharing transcripts of his LEGO/Logo work with children. Edith Ackermann, Idit Harel, Fred Martin, and Andee Rubin provided helpful comments on earlier drafts of this paper.

Our research on LEGO/Logo is supported by grants from the LEGO Group and from the National Science Foundation (Grants 851031-0195, MDR-8751190, and TPE-8850449). LEGO Systems Inc. markets a version of LEGO/Logo under the product name *LEGO TC logo.*

Portions of this paper previously appeared (under the title "LEGO, Logo, and Design") in *Children's Learning Environments* (vol. 5, no. 4), winter 1988.

References

Dewey, J. (1938). *Experience and Education.* Macmillan Publishing Co. New York.

diSessa, A. (1986). "Artificial Worlds and Real Experience," *Instructional Science,* vol. 14, pp. 207-227.

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas.* Basic Books. New York.

Papert, S. (1986). "Constructionism: A New Opportunity for Elementary Science Education," a proposal to the National Science Foundation.

Resnick, M., S. Ocko, and S. Papert (1988). "LEGO, Logo, and Design," *Children's Environments Quarterly,* vol. 5, no. 4.

Simon, H. (1969). *The Sciences of the Artificial.* MIT Press. Cambridge.