Published in the Proceedings of CHI 2000, April 1-6, 2000, ACM Press, ©2000 ACM

curlybot: Designing a New Class of Computational Toys

Phil Frei, Victor Su, Bakhtiar Mikhak*, and Hiroshi Ishii
Tangible Media Group
*Epistemology and Learning Group
MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139
+1 617.253.9401
{frei, vsu, mikhak, ishii}@media.mit.edu

ABSTRACT

We introduce an educational toy, called *curlybot*, as the basis for a new class of toys aimed at children in their early stages of development – ages four and up. *curlybot* is an autonomous two-wheeled vehicle with embedded electronics that can record how it has been moved on any flat surface and then play back that motion accurately and repeatedly. Children can use *curlybot* to develop intuitions for advanced mathematical and computational concepts, like differential geometry, through play away from a traditional computer.

In our preliminary studies, we found that children learn to use *curlybot* quickly. They readily establish an affective and body syntonic connection with *curlybot,* because of its ability to remember all of the intricacies of their original gesture; every pause, acceleration, and even the shaking in their hand is recorded. Programming by example in this context makes the educational ideas implicit in the design of *curlybot* accessible to young children.

Keywords

Education, learning, children, tangible interface, toy

INTRODUCTION

The role of physical objects in the development of young children has been studied extensively in the past. In particular, it has been shown that a careful choice of materials can enhance children's learning. A particularly notable example of such materials is Friedrich Froebel's collection of twenty physical objects (so called "gifts"), each designed with the purpose of making a particular concept accessible to and manipulable by children [5]. The presence of objects inspired by Froebel in almost all kindergartens today is a reflection of their recognized value in the development of young children.

Figure 1: Three palm-sized *curlybots* (each with a large record/playback button and a small indicator light).

Most recently, Mitchel Resnick and his Lifelong Kindergarten Group at the MIT Media Laboratory have introduced a collection of digital manipulatives that builds on Froebel's work, taking full advantage of computational ideas and resources not available until recently [12,14]. Much like Froebel's gifts, these tools attempt to make new domains of knowledge accessible to children.

In this paper, we contribute to this initiative a new class of computational toys that is aimed at children as young as four. *curlybot*, the first instantiation and the basis of this class of toys, is a two-wheeled toy that can record and play back physical motion replicating every intricacy of the original motion. It is a smooth, easily graspable curved object with a button and an LED for indicating whether the device is in record (red) or playback (green) mode. To record a gesture, a child presses the button and moves *curlybot* through a desired path. A child presses the button a second time, to stop recording and begin playback of the recorded gesture. The playback mode repeats the gesture indefinitely until the button is pressed again. Because of the simplicity of the interface, children quickly learn to create intricate gestures with *curlybot*, which they can refine through an iterative process.

This version of *curlybot* also has a mode called "boomerang," in which a *curlybot* will move backwards through its path and then forward again. Holding the button when the toy is turned on activates this mode. *curlybot* also has a pen attachment to create drawing of gestures (see Figure 2).

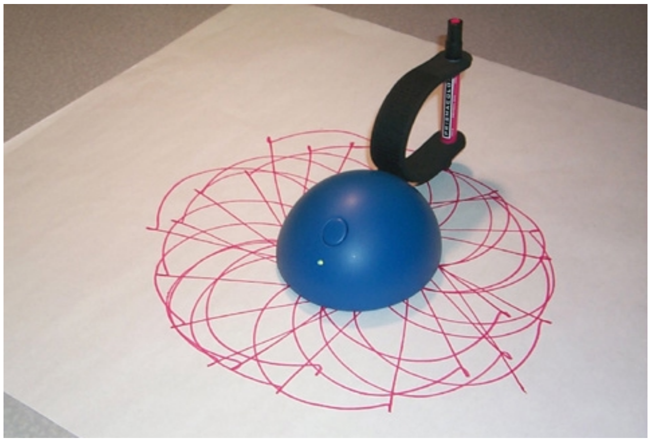

Figure 2: *curlybot* with a pen attachment.

We believe that *curlybot* can play a significant role in mathematics education research. Therefore, a part of this paper will be devoted to the discussion of the educational issues that *curlybot* is designed to address. We will propose a set of possible play scenarios for *curlybot* to highlight different educational ideas, which extend *curlybot* into a class of new toys that support multiple forms of play. Finally, we will review the user testing and present the design and implementation of the current system.

MOTIVATION

Many of the computational environments designed for children have been thus far limited to activities on the computer screen. One notable example that has enjoyed great recognition in and out of the classroom is graphical Logo. The main computational object in Logo is a turtle, whose heading and trajectory can be controlled by simple programs written by children. Graphical Logo was inspired by a small robot (about one cubic foot in size) built at the MIT Artificial Intelligence Laboratory by Seymour Papert and collaborators. This robot, called the Floor Turtle, was quite heavy and tethered to a mainframe computer. By typing commands at a terminal, children controlled the turtle and its pen to draw geometric patterns on large sheets of paper on the floor. As Graphical Logo was developed and soon became widely used, the Foor Turtle was put on hold.

In the 1980's, Fred Martin, Seymour Papert and Mitchel Resnick resurrected this work at the MIT Media Lab by building computation and programmability into the familiar LEGO bricks. Children could build these Programmable Bricks into their robots and program them to bring their creations to life. The most recent member of the Programmable Brick family is the Cricket, which encapsulates the core functionalities of the previous generation into a much smaller package and makes the system expandable through a unique bus structure. The Programmable Brick inspired the LEGO Mindstorms Robotic Invention System [10].

Robots built with the Programmable Bricks and Crickets are currently programmed in text-based or graphical programming languages that are dialects of Logo. Research has shown that children as young as ten years can successfully use Programmable Bricks and traditional construction material to build and program their own robots to exhibit the behavior they are looking for. Extending these types of activities to younger children is an active area of research [13].

The design of *curlybot* is also inspired by the natural and expressive quality of Golan Levin's gesture-based animation environment system called Curly [8], which builds on Scott Snibbe's Motion Phone system [19]. These systems capture the gestures of the computer mouse on the screen and replay it graphically.

Execution/Output \ Programming/Input	Digital	Physical
Digital	Turtle Graphics	Motion Phone Curly
Physical	Floor Turtle LEGO/Logo Programmable Bricks Crickets LEGO Mindstorms BigTrak	curlybot

Table 1: Input vs. Output Interaction Space.

Table 1 summarizes the differences in the ways in which children interact with the various design and expression media we have discussed so far. This parameterization of possible modes of programming and interacting with computational media highlights *curlybot's* significance, including the coincidence of the input and the output space.

INTERACTION & DISCUSSION

We envision children having several basic interactions with the *curlybot* family of toys. The current design of *curlybot* can be extended in multiple directions. Some of the activities in this section rely on augmentations to the current system that we will discuss in conjunction with the various activities.

Repetition – How do you keep the toy repeating a gesture while not falling off a table? A child, in this case, would learn to create repetitive patterns that as a rule would end up at the origin or circle around a central point. Through this direct manipulation, a child can learn many lessons by just playing and experimenting with movement, spatialization and repetition. Another example is a child trying to create a star with three gestures. This activity introduces a child to the idea of building complex shapes by combining simpler elements. A child is also exploring computational and mathematical ideas, like loops and vectors. To create the star, you have to be concerned with elements of a vector, such as point of origin, direction, and magnitude. When *curlybot* loops the recorded vector, it is critical to start and finish with correct orientation, not just position. A pen can be attached to *curlybot* to leave a trail of its path and make the visualization of more complex pattern easier (see Figure 3).

Figure 3: Examples of pen drawings.

Making it possible to record and play back whether the pen is up or down would allow for a broader range of designs that include discontinuous lines, like dotted or dashed lines.

Pen Position - The use of pens introduces additional mathematical concepts, since the pen can be placed in different locations relative to the wheels of the vehicle. For example, a *curlybot* asked to move forward and turn 90 degrees, will create a square, if the pen is placed exactly between the two wheels (see first pattern in Figure 4). However, a different pattern will emerge, if the pen is placed farther from the center. This should be contrasted with the graphical turtle, which is assumed to be a point-like object with its pen located at its center. *curlybot* allows for more surprising patterns to emerge which encourages a child to think about the distinction between point-like and extended objects. A child might not mathematically understand the concept, but will have at least developed intuitions for relative position and motion of points.

If one adds an additional degree of freedom and had the pen move independently in a circle around *curlybot*, one can create more complex patterns that begin to mimic orbital patterns. If we then moved *curlybot* in a circle and had the pen move at a higher frequency around *curlybot*, one creates the orbital pattern of the moon relative to the sun.

Conditional Behavior – Additional sensors could be added to *curlybot*, like bump and light sensors, in order to program conditional behavior. For instance, one could teach *curlybot* to move forward and it would then drive straight until hitting a wall with one of its bump sensors. At that point, the toy would stop moving. The LED on the device would turn yellow, prompting the user to record a sequence in response to hitting the wall. One could then record, going backwards a little and turning, which would now be *curlybot's* standard response to hitting an obstacle with that particular sensor. This type of conditional programming would allow *curlybot* to respond to its environment instead of simply playing back a recorded gesture allowing *curlybot* to act as an autonomous creature with complex behavior.

This type of behavior is the same as that of creatures made with Programmable Bricks. However, *curlybots* are programmed by example rather than using traditional programming. Nonetheless, a child can still learn about "if" and "while" statements.

Recording Primitives – If one records several sequences and stores them on a computer under different names, like circle, box, and line, they could later be used as procedures in a programming language such as Logo. Separate gestures could then be combined together in a computer program and sent back out to *curlybot*. This added functionality leverages the simplicity of physical programming and gestural output with the added flexibility of a computer program. This is a concrete example of procedural abstraction.

Gesture and Narrative - Since *curlybot* captures not only the trajectory of movement but also velocity and acceleration, it can be used to express gesture. For example, a child could record a nervous shaking, and *curlybot* would do just that. This gestural expression is also useful in playing out a story with the toy. It is common for children to act out stories with toys, but in this case the toy could boomerang back through all the obstacles and start replaying the interaction physically - pausing and accelerating in all the right places. One could even add audio recording and playback to the device and synchronize what the children say with their movements [16]. The child could learn aspects of storytelling and gesture by watching his or her own actions from the point of view of an observer.

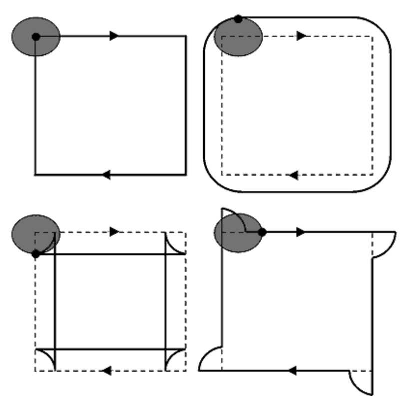

Figure 4: Four different pen positions and their resultant trails on the same *curlybot* pattern.

Synchronization – Two or more *curlybots* could be synchronized to create a medium for haptic communication, like the inTouch [4] or HandJive [6], or a display of remote physical activity, like Ambient Displays [7].

Music – One could map different gestures created with *curlybot* to musical sounds and have them loop like *curlybot's* physical motion in a rhythmic pattern. Alternatively, one could affect a musical piece that is played.

Trading – The exchange of digital information is a very rich area of research. *curlybot* supports the exchange of its information because its memory is physically removable and can be used to save a session or exchange it with someone else's *curlybot*. The memory could also be exchanged with a personal computer, where the recorded path could be displayed, or potentially altered and resaved to the memory. The file could be sent to distant friends to be played on their *curlybots*. Some interesting issues arise in the representation of the information, since we are not only recording a trajectory, but also velocity and acceleration. There could be some interesting challenges in the visualization and editing of this information.

The exchange could also happen without a physical exchange of memory, but rather through infrared (IR) or radio frequency (RF). One could have one *curlybot* teach another *curlybot* an interesting gesture and these could be passed on and saved It would be interesting to see how a particular pattern spreads and to examine which gestures people felt compelled to pass on to others. One could also introduce evolutionary ideas, involving the progressive alteration of patterns over time via an exchange with different patterns [2,3].

Different Personalities - One can change the variables in the control algorithm to create different "personalities" in the *curlybots.* For example, a *curlybot* could be designed to be better at fast motions, while another could be designed to be better at slow motions. Creating these distinct *curlybots* gives them personality outside the recorded gesture, making them individual characters which children will be drawn to in different ways.

Editing - There could be other forms of input, like electric field sensing, that could be used to change the motion as it is playing it back

EDUCATIONAL IMPLICATIONS

In this section, we will discuss the complementary educational opportunities afforded by the *curlybot* family of toys in light of its potential to:

- serve as objects-to-think-with [11]
- make new domains of knowledge accessible or old domains of knowledge approachable in new ways
- support multiple learning styles [15]

curlybot as an object-to-think-with

curlybot's physical form, size and weight makes it a natural extension of the hand which a child can program by example. A child can map ideas from his or her mind directly into a clear physical instantiation of the ideas. The process and validity of the execution is transparent because the motion involved in the act of programming is bodily syntonic. The immediate feedback from the observed behavior of the robot allows children to examine and reflect on their initial mental models with respect to the outcomes they observe and gives them a chance to debug and extend their thinking.

In *Mindstorms*, Papert eloquently describes the significance of programming as a tool for thinking about one's own thinking [11]. The very process of externalizing models and concepts in ones mind into the physical world allows for the critical evaluation of the validity of the models by oneself and others against easily understandable physical behavior. In turn, the external instantiation of an idea can be internalized again to modify the initial models. *curlybot* takes advantage of the rich educational opportunities afforded by creating and supporting such internalization/externalization feedback loops.

curlybot and new domains of knowledge

curlybot makes the core ideas in Logo accessible to even younger children. *curlybot* can provide a tangible way of exploring many important ideas that have been studied extensively within the Logo community. For example, moving forward a little and turning a little will result in a circle, if one repeats it over and over again. This will result in a more even circle than if the child tried to create the circle out of a single gesture. This is a concrete instantiation of the idea of differential calculus as well as local representation of a circle.

In addition to differential calculus or local and intrinsic representation of curves, *curlybot* could be used as a tool to gain intuitions for turtle geometry [1], Aristotelian and Newtonian physics [12], and the law of large numbers and probability [19], to name a few. Many of these topics are ordinarily considered too advanced for children, but interacting with carefully designed objects can make this material accessible to them.

curlybot and multiple styles of play and learning

An equally important component of any powerful learning experience is the affective quality of the relationship between the learner and the material. Studies have shown that children's learning and play patterns can be divided into two overlapping categories, namely patterners and dramatists [17]. In the design of *curlybot*, we were conscious of supporting both forms of play. Whether a child is a planner or a dramatist, he or she will connect to the same mathematical ideas but in ways that are more natural for them (see section on user interaction for more examples).

curlybot can engage children who are more artistic and expressive. The entry point into mathematics for these children is through their artistic involvement with a tool and a medium. In this case, the critical feature of *curlybot* is that it lives up to children's expressive expectations. This allows children to make a strong affective, as well as intellectual, connection to *curlybot*.

Papert makes a strong argument for the importance of this last idea in his book *Mindstorms*. He begins by explaining how gears taught him "advanced" mathematical ideas as a child, but then claims that giving sets of gears to children will not necessarily result in the same learning experience for most of them. The success is in part due to the child's personal attachment to the gears - Papert "fell in love" with his gears. He could project himself into the gears and "be the gear," which is what "gives the gear the power to carry powerful mathematics into the mind." If a child is not completely engrossed in playing with a toy, they will not learn very much from it [11].

USER INTERACTION

We began by allowing several hundred adults to play with *curlybot* and found that many of them discovered new gestures and patterns that we had not anticipated. This, of course, was a promising result, since our hope was to design an open-ended toy that would continue to be interesting over time.

Figure 5: Child playing with *curlybot* on Plexiglas.

In particular, it was interesting to see people take advantage of the fact that *curlybot* records every pause one makes. In one case, someone had *curlybot* do nothing for a long time and then shake around. This resulted in an interesting behavior during playback: *curlybot* would appear inactive or off, but then surprise the audience by suddenly starting to shake. Another user recorded a pause, a shake forward and back, a pause, and then a shake from side to side. When playing back, he asked *curlybot* if it liked him, and it moved forward and back. He then asked if it liked his friend, and it shook from side to side. By having others play with *curlybot*, we discovered a real satisfaction in learning a new behavior or pattern.

First Study with Children

The first study was conducted at the Science Museum in Boston, Massachusetts. Though this was not a completely random cross section of children, it easily provided us with a large group (81) of children for initial tests. The Science Museum is a good environment for making observations, since the children are prepared to play with things and generally do not notice when someone is observing them.

In the entrance to the Discovery Center of the Museum, we set up a 3'x 4' piece of Plexiglas to clearly demarcate the play space. This was also done to observe if the children would learn how to keep *curlybot* in that space. The play was forced to remain on the Plexiglas, since we used a version of *curlybot* that was not designed to run on the surrounding carpet.

Very little instruction was given to the children, in order to learn how effective the interface was from the start. We then were interested in monitoring what children did with *curlybot*. Did children figure out how to keep *curlybot* from running outside of the demarcated area? Were they more interested in geometric designs, gestures, or narratives? How long did it take them to figure it out? Can we generalize the responses of different age groups? Is there an age where children cannot interact with the device at all? It should be noted that our results are based on qualitative observations, and subjective categorization. These results are nonetheless interesting, because they provide us with a rough guide for further study.

Most of the children knew what to do with *curlybot* by observing how others had used it. If they did not, we would ask one of the other children to explain it to them. Through this, we were able to observe if they had learned something beyond the basic functionality of how to record and play. Out of the twenty-two children who were asked to explain how to use the toy to someone else, only three of them described how to keep *curlybot* on the platform in addition to explaining the basic functionality.

About a quarter of the children (21 out of 81), explicitly created geometric shapes. Four children did what we considered to be explicitly gestural recordings, while the rest did narrative recordings. It was difficult to draw lines between the different interactions, since there was some overlap between the categories. One ten-year-old girl, for example, recorded a beautiful geometric piece after observing four boys of her age record strictly geometric shapes. However, unlike the boys, her geometric shapes had accelerations and pauses, which created a more gestural pattern. This made us categorize her actions as gestural rather than geometric, even though she was also very successful at keeping *curlybot* on the platform through a geometric pattern. It is interesting to note that the boys were impressed and tried to create some more gestural patterns after her performance. This also shows that a child can be affected by another child's interaction with *curlybot*. Our results are heavily affected by this fact, since we were not working in a controlled environment where children were isolated from one another while playing with *curlybot*.

We hoped to see trends in play between the different age groups, however the only conclusive results we found were that children under the age of four generally could not meaningfully interact with *curlybot*. We also thought that older children might not learn much from the interaction, but that did not seem to be the case. Older children spent just as much time as younger ones trying to figure out how to design a pattern that would stay on the platform.

It was interesting to observe that the children had a tendency to make large and fast gestures with *curlybot*. This caused two problems. One, because there was a constrained play area, large motions, that did not end exactly where they began, made *curlybot* fall off the platform. Two, this version of *curlybot* was not designed to reproduce fast motions as accurately as slow ones and, as a result, *curlybot* did not repeat geometric shapes perfectly. Overall, the children were not concerned with these problems and continued to play with *curlybot* anyway. For future tests, though, we will redesign the control algorithm.

It usually was not possible to have children perform specific tasks given the informal environment of the study. However, there was one seven-year-old girl that played with *curlybot* for an extended period of time and accepted our challenge to create a few geometric shapes out of their most basic elements. We found that she needed us to provide an example before being able to create the shapes on her own. We showed her how to create a square and let her try it on her own. When we asked her to create a circle, she started by designing it with very large arcs. She needed additional help to understand that a circle could be created from a very small segment. Later on, the same girl came back, and asked if she could try a shape she had been thinking about. We were pleased to see that she continued to process her new knowledge about shapes even outside the play area. *curlybot* appears to have become an object-to-think-with for her.

Though this user test was not conclusive, it confirmed that *curlybot* is fun for children and that our questions were indeed relevant in view of the children's interactions with the toy. In particular, we would like to present children with specific design challenges, like creating a complex pattern out of simple elements and study how they would perform and their thinking process.

IMPLEMENTATION

Current Implementation

The *curlybot*'s two wheels have independent drive and sensing capabilities that are controlled by a microprocessor. Mechanically, the toy consists of two 10 Watt Maxon motors with Hewlett-Packard Optical encoders. They are mounted on the bottom of *curlybot* in such a way that, after gearing the torque up 4:1, the shafts of both wheels are co-linear. This allows it to not only move forward and back, but also rotate freely about its center. This is also the most compact design that allows the device to easily fit in the user's hand. The physical configuration also simplifies what needs to be recorded. If both motors are moving forward, the device is moving forward. If they are moving in opposite directions, then the device is turning.

The 10 Watt motors are very efficient and power is not lost in heat dissipation. The use of these large motors gives us additional mass, which is useful in creating sufficient friction for the drive wheels. In this way, the user can feel resistance when they push against the direction of the wheels. Also, the additional weight creates a good inertia for play.

A 20MHz Microchip microprocessor with built-in pulse width modulation controls the motors. The encoders available to us had 500 counts/revolution. Because of the gearing, the resolution of the wheel is 2000 counts/revolution. If *curlybot* is moving quickly, the encoder interrupts the microprocessor continuously, which does not allow other processes to be run. To overcome this, we divide the encoder information by four using a counter, so that the resolution of the wheel is only 500 counts/revolution.

Figure 6: Top and Bottom of *curlybot*.

The encoder information is stored on a separate 32 kilobyte memory chip (256 kilobits) at a rate of 100Hz. At this rate, we can record the encoder information of both motors for about two and a half minutes. The device currently runs on six AAA batteries – four for the motors and two for the circuit board.

Originally, we used two 9 Volt batteries in parallel for the whole system, but there were two problems. One, the capacity of 9 Volt batteries is much less than that of AAA, so a *curlybot* would not run continuously for more than two hours. Second, when the motors draw a lot of current, the voltage for the circuit board drops below 5 Volts and the circuit resets. We also originally used a one-megabit serial eeprom memory chip, since we were not sure with what frequency we wanted to record. When we finally decided that 100Hz would be enough, this memory chip gave us about ten minutes of recording time, which is much more than what we needed. We then switched to our current eight-pin 256 kilobit eeprom memory chip that can be easily removed from the board and replaced with any other 8 pin eeprom. It also has a fraction of the leads, since one reads and writes to it serially.

The motor is run on pulse width modulation with feedback only from the encoder. The performance of the playback could be improved by monitoring the current feedback from the motor.

To record, the user presses a button that lights up a red indicator LED. When the user is done recording a sequence, the button is pressed again and the indicator LED turns green. At this point, the processor runs a PID control function that calculates the force that the motors need to exert to reach the recorded position. The processor compares its current position (from the encoder) to the desired position (from the memory) and then applies the necessary force to move from one to the other. When the button is pressed again, the indicator LED turns off and *curlybot* is in neutral mode. Here it is free to roll around and nothing is recorded or played back. The sequence can be started again by pressing the button one more time.

We can also switch *curlybot* into boomerang mode, by pressing the button while turning the device on. In this mode, the toy boomerangs back through its recorded path to its starting position, where it then begins to repeat the motion again.

Another Implementation

In order to test some of our other interface ideas, we decided to design another version of *curlybot*. First, we added a two-button interface with separate record and playback buttons. This allows users to re-record a motion without playing it back or, likewise, to stop playing a motion and then start again without re-recording. We have also explored using a double-click on the single button interface to click over the record or playback mode. This provides the additional functionality of the two-button interface without making it more confusing for novice users.

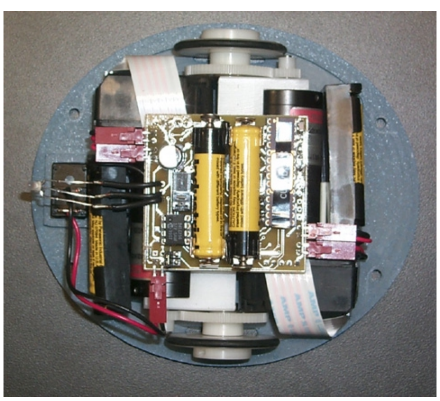

Figure 7: Inside *curlybot* (top view).

We have also reduced the size of *curlybot* to something smaller than a computer mouse. This version uses 1 Watt Maxon motors that are about the size of a AAA battery, including a 12 count/revolution encoder and 4:1 gearhead. Though the resolution of the encoder is lower, we still managed to maintain about the same resolution on the wheel circumference. To keep the toy small, we used two AAA batteries to run both the circuit board and the motors, even though we knew we could run into problems with high current draws. The main problem with this prototype was that, because it was lighter and smaller, the wheels' traction was not enough when a user pushed against the direction the wheels turn.

RELATED WORK

The Epistemology and Learning Group at the Media Lab, as mentioned in the Motivation section, has done very closely related work for many years, spearheaded by Seymour Papert, Mitchel Resnick and Fred Martin. This work includes Logo, LEGO/Logo, Programmable Bricks, Crickets, and LEGO Mindstorms Robotic Invention System. The ideas for trading information between *curlybots*, mentioned earlier, are based on the research of Rick Borovoy, such as his MemeTags and Dance Craze Buggies [2,3].

The work of Kimiko Roykai and Justine Cassell called StoryMat is about creating a space that encourages children to tell stories with a plush toy and later have them replayed. The replay is not in physical form, but occurs with a moving projection of the toy on the StoryMat accompanied by the recorded audio [16].

Microsoft's ActiMates Barney, like *curlybot*, attracts the child's attention by being a character that exists in the child's physical rather than virtual space. One of the major differences, though, is that Barney is a story-based toy. This means that the child's interactions with Barney are limited by a preprogrammed or uploaded set of stories. *curlybot* on the other hand, invites the child to discover by playing. Instead of being told a story or being given a specific task, the child learns through teaching *curlybot* and exploring the results. Because this interaction is more complex, Barney is still easier to use for very young children [20].

In manufacturing, to save time programming robotic arms in assembly lines, the robot is physically given end points for its trajectory and is then allowed to calculate the optimal path. If there are obstacles for the robot arm to avoid, extra points are added to create the desired trajectory. Like *curlybot*, this is an example of physical programming.

Similarly, in robotic artificial intelligence, researchers have for some time used techniques of programming a robot by recording the motion it should perform. For example, with the help of a human hand, one can quickly program the many degrees of freedom in three dimensions of a robotic arm picking up a cup.

FUTURE WORK & CONCLUSION

The first and most important next step is to perform controlled user studies with children in order to determine what types of things children learn from their interaction with *curlybot*. This means that we will need to give children a longer time to play with *curlybot,* so that they have time to explore its full range of possibilities. We will make systematic observations of this interaction. We will also present children with specific design challenges, in order to determine what they are learning and how they are thinking about accomplishing the goals of the challenge. We will follow the activities with interviews. A longer term study would be needed to reveal if and to what extent interacting with *curlybot* prepares children for working in text-based or graphical programming environments, such as Logo. These types of studies are much more challenging, since it is difficult to isolate the contributions from a specific source to a child's future abilities.

Currently, we are focusing on the implementation of the augmentations mentioned in the Interaction & Discussion section. These would provide different computational and mathematical concepts for children to explore, which could confirm *curlybot* as a toy capable of supporting multiple learning and play styles. Furthermore, these new implementations may lead us to discover new directions for this research.

In conclusion, our preliminary results show that *curlybot* succeeds in engaging children ages four and above to play around with advanced mathematical and computational concepts (previously learned at a later age and often with the aid of a traditional computer) in a much more fluid and expressive fashion. *curlybot* is an introductory tool, much like Logo, that can help build a child's basic mathematical intuition by engaging them in genuine mathematical activities. Once this basic understanding has been established, children will have an easier time moving into computer programming and formal mathematics as they get older. The example interactions presented in the paper, position *curlybot* as the basis for an entire class of computationally enhanced educational toys, which we are actively designing.

ACKNOWLEDGMENTS

I would like to thank Brad Niven of Interval Reseach Corporation, Ali Malazek, Megan Galbraith, Rujira Hongladaromp and all the other members of the Tangible Media Group at the Media Lab who have contributed their ideas and time to this project. This project has been supported by the MIT Media Lab's Things That Think consortium and Interval Research Corporation.

REFERENCES

1. Abelson H., and diSessa A. (1981). *Turtle Geometry*. MIT Press.
2. Borovoy R., and Martin F. (1999). Tradable Bits. <http://el.www.media.mit.edu/people/borovoy/cars/>.
3. Borovoy R., Martin F., Vemuri S., Resnick M., Silverman B., and Hancock C. (1998). Meme Tags and Community Mirrors: Moving from Conferences to Collaboration. *Proceedings CSCW '98*, ACM Press, 159-168.
4. Brave, S., and Dahley, D (1997). inTouch: a medium for haptic interpersonal communication. *Extended Abstracts of CHI'97*, ACM Press, 363-364.
5. Brosterman, N. (1997). *Inventing Kindergarten*. New York, Harry N. Adams Inc.
6. Fogg, B.J., Cutler, L., Arnold, P., and Eisback, C. (1998) HandJive: A device for interpersonal haptic entertainment. *Proceedings of CHI'98*, ACM Press, 57-64.
7. Ishii, H. and Ullmer, B. (1997). Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. *Proceedings of CHI'97*, ACM Press, 234-241.
8. Levin, Golan. Curly. <http://acg.media.mit.edu/people/golan/curly/>.
9. Kafai Y., and Resnick, M., eds. (1996). C*onstructionism in Practice:* Designing, Thinking, and Learning in a Digital World. Mahwah, NJ, Lawrence Erlbaum.
10. Martin F., Mikhak B., Resnick M., Silverman B., and Berg R. (1999). To Mindstorms and Beyond: Evolution of a Construction Kit for Magical Machines. *Robots for Kids* edited by Alison Druin and James Hendler, Morgan Kaufmann Publishers, Inc.
11. Papert, Seymour (1980). *Mindstorms*: Children Computers and Powerful Ideas. BasicBooks.
12. Resnick, M. (1998). Technologies for Lifelong Kindergarten. *Educational Technology Research and Development*, vol. 46, no. 4.
13. Resnick, M., Eisenberg, M., Berg, R., and Martin, F. (1999). Learning with Digital Manipulatives: A New Generation of Froebel Gifts for Exploring "Advanced" Mathematical and Scientific Concepts. Research proposal, May 1999.
14. Resnik A., Martin F., Berg R., Borovoy R., Colella V., Kramer K., Silverman B (1998). Digital Manipulatives: New Toys to Think With. Paper Session, *Proceedings of CHI'98*, ACM Press, 281-287.
15. Turkle, S., and Papert, S. (1990). Epistemological Pluralism. *Signs 16*, 1, 128-157.
16. Ryokai K., and Cassell J. (1999). StoryMat: A Play Space for Collaborative Storytelling. *Extended Abstracts of CHI'99*, ACM, 201.
17. Shotwell, J., Wolf, D., and Gardner, H. (1979). Exploring Early Symbolization. In B. Sutton-Smith (ed.), *Play and Learning*.
18. Silverman B., and Tempel M. (1991). Fuzzy Logo. Logo Foundation Memo. <http://el.www.media.mit.edu/groups/logo-foundation/Publications/Fuzzy-Logo.html>.
19. Snibbe, Scott (1995). Motion Phone. Interactive Communities, SIGGRAPH '95.
20. Strommen, Eric (1999). When the Interface is a Talking Dinosaur: Learning Across Media with ActiMates Barney. *Proceedings of CHI'99*, ACM Press, 288-295.