

## **Robotic Technologies: When Parents Put Their Learning Ahead of their Child's**

LAURA BEALS AND MARINA BERS

*Tufts University, USA*

[laura.beals@tufts.edu](mailto:laura.beals@tufts.edu)

[marina.bers@tufts.edu](mailto:marina.bers@tufts.edu)

New technologies are slowly making it into the early childhood classroom; however, many families expose their children to these technologies before they encounter them in the formal school setting. Thus, many parents serve as the first teachers of technological literacy to their children. Or, as it is more frequent, parents and children learn together new computational skills. What are the complexities hidden behind this learning process in which both adults and children are learning at the same time? In this article we present results from a study in which, during five weekend sessions, 17 parent-child dyads and 20 children were taught to use robotic programmable Lego™ bricks to create their own meaningful final projects involving both programming and building components. In the study, a significant difference was found between building and programming aspects of final projects between the children-only projects and the parent-child projects. This suggests that Vygotsky's zone of proximal development played an important role, but it is argued that the children in the parent-child groups did not learn as much as the children in the children-only groups, as the parents were too involved in their own learning and did not tailor their instruction to their children at a level appropriate for the children to understand.

### **Introduction**

In the past decade, there has been an overwhelming surge of technology that is significantly influencing daily life (e.g, cell phones, personal computers, the Internet). Children are becoming more exposed to technology at earlier and earlier ages – many elementary school children now carry cell

phones and recent government surveys show that in 2003 over sixty percent of households in the United States have a personal computer and over fifty percent have Internet access (U.S. Census Bureau, 2005). Young children have the almost uncanny ability to pick-up how to use technology with ease, while their parents often lag behind on the learning curve (Papert, 1996). Thus the traditional "parents teach children" mentality is being overturned – when it comes to technology, it is often the children who teach the adults. Although this statement is frequently made, what is the theoretical basis for this? What are the complexities hidden behind this simple statement? Is it really true that children are teaching their parents? What is the relationship between developmental aspects and complexities of the technologies been used? And what happens when both parents and children together learn a new technological skill? What are the dynamics of a teaching-learning relationship in a parent-child dyad?

This article explores some of these questions by presenting results from a series of intensive five-session workshops conducted as part of Project Inter-Actions. (Bers, New, & Boudreau, 2004). This project was designed to examine: (a) the interactions between children and parents as they both learn a new technology and as they work together to create personally meaningful projects; (b) the interactions between an abstract technology, such as a computer program, and a concrete technology, such as Lego bricks; (c) the interactions between the technological ability of young children and the developmental appropriateness of technology for young children; and (d) the interactions between technology, art, and culture. Because the project would be looking at these four types of interactions, it was dubbed Project Inter-Actions (PI). In this article we examine the interactions between parents and children as they both learn robotics technology to design and implement personally meaningful projects.

Project Inter-Actions has its roots in project Con-science, held in a Jewish community school located in Buenos Aires, Argentina. During the project, which was held during the Jewish High Holidays, families worked together to create *technological prayers* that used robotic Lego bricks. Each of the families then had an opportunity to exhibit and demonstrate their projects during an open house held at the local synagogue, attended by members of the community (Bers & Urrea, 2000).

In Con-science, the research goal was to engage families to work together to explore their own heritage. However, while conducting this research, interesting teaching and learning dynamics were observed when both children and parents were faced with the need to learn a new technology. Some of these dynamics were due to the pre-adolescent age of the children involved in Con-science and therefore went beyond the scope of our interest because they pertained to developmental issues of control and separation. However, interesting anecdotal data was collected regarding parents and

children learning together. In order to explore this in more depth, it was decided to conduct Project Inter-Actions with younger children – first and second graders (six and seven years old). Developmentally, children at this age are happy and look forward to working with their parents and relationships are much simpler (Furman & Buhrmester, 1992).

Based on the experiences during Con-science, we had two guiding research questions for Project Inter-Actions: What happens when parents and children learn together a technology (such as robotic Lego bricks) that is new for both of them? and what are the advantages and disadvantages of this type of learning? We believe that the issues raised with these questions are important in understanding how to create learning environments that go beyond traditional schooling. Much research has shown the difficulties of introducing new technologies in schools, in particular with younger children (Cuban, 2001). Most of the obstacles are not due to the limitations of the technology, the children, or the teachers, but rather because of the logistical set-up of working in short-blocks of time with one teacher per 20 or more children (Rogers & Portsmore, 2004; Erwin, Cyr, & Rogers, 2000). Understanding the type of learning that can happen when parents and children engage in learning together is a first step to envision alternatives to the teaching and learning of educational technologies, either in homes, museums, or community-based centers.

In this article, we will first discuss the guiding educational philosophies and background of the project. Next, we will describe the methodology of the study, including a description of the study organization, the participants, the materials, the instructors, and the projects created. Finally, we will discuss our method of analyzing how parents and children worked together, present results from the experiences, and discuss the implications of our findings for learning at home or in after-school settings.

## **Background of Study**

### ***Guiding Educational Philosophies***

Project Inter-Actions was developed with three major educational philosophies and concepts in mind: (a) the philosophy of constructionism, (b) the concept of the zone of proximal development, and (c) the concept of peer learning environments. In the following subsections we address how each of these theoretical models informed our own work.

### ***Papert's Constructionism***

Seymour Papert is credited with developing the idea of constructionism, based on Piaget's theory of constructivism (Papert, 1980). Following Piaget's stage theory of cognitive development, children participating in this research would be expected to be in the preoperational stage. This stage is marked by the development of symbolic symbols, including speech, or the concrete

operational stage, “distinctive for its focus on the empirical reality of experience, a preoccupation with the facts and how they can be documented, and an emerging set of logical operations that permit hierarchies to be formed and classes to be established” (Feldman, 2004, p. 207). The adults, however, would be expected to be in the formal operational stage, identified by the development of hypothetical and abstract thought. The children in this study, in particular those who had not yet reached the concrete operational stage of development, may have had difficulty sequencing and constructing the robotic models, which in comparison to the adults’ abilities to complete these two tasks, further created barriers to effective learning with their children.

Papert, building upon the work of Piaget, described constructionism as “built on the assumption that children will do best by finding (‘fishing’) for themselves the specific knowledge that they need,” and he adds, “The kind of knowledge that children most need is the knowledge that will help them get more knowledge” (Papert, 1993, p. 139). He also encourages learning that has a *concrete* product that can be displayed and shown to others, and thus “constructions in the world” support “constructions in the mind.”

Lego kits are especially ideal as support for constructionism because students can explore abstract thoughts, ideas, and designs with concrete materials. Bers identified four pillars of educational experiences designed within a *constructionist framework* (Bers, Ponte, Juelich, Viera, & Schenker, 2002): (a) the potential of technological environments to help learners learn by doing, by actively inquiring, and by playing; (b) the importance of objects for supporting the development of concrete ways of thinking and learning about abstract phenomena; (c) the need for powerful ideas that span across different areas of the curriculum; and (d) the premium of self-reflection which engages learners in meta-cognition. These four pillars informed the design of the workshops held within Project Inter-Actions:

1. *The potential of technological environments to help learners learn by doing, by actively inquiring, and by playing.* The interaction with technological materials provides learners with the opportunity to design and make meaningful projects to share with a community. In Project Inter-Actions, the combination of the Lego robotic bricks, the programming software, and the art materials afforded unlimited options for designing a project that had personal importance that was shared with other participants throughout the project as well as family members during the final open house.
2. *The importance of objects for supporting the development of concrete ways of thinking and learning about abstract phenomena.* In this context, computers and the ability to program them, acquire a salient role as powerful tools to design, create, and manipulate objects in both the real and the virtual world. The RCX, the main

component of the Lego robotics kits (it will be explained in more detail below), provides a powerful platform because it combines the concrete Lego blocks with programming abilities, allowing people to truly support constructions in the mind with constructions in the real world.

3. *The need for powerful ideas.* Powerful ideas can be domain specific (ideas that form the core historical knowledge of a discipline such as the concept of zero or the process of addition) or they can cut across domains (such as problem solving or taking perspective). Research done within the constructionist philosophy suggests that best curriculum frameworks are organized around powerful ideas, and not only around the content matter of a discipline. What is the difference between content matter and powerful ideas? Powerful ideas afford individuals new ways of thinking, new ways of putting knowledge to use, and new ways of making personal and epistemological connections with other domains of knowledge (Papert, 1980). They go beyond specific content or skills and they provide a way to think of that content in context. Technologies can be a powerful carrier of new ideas, and in this sense, they can become an agent of educational change.
4. *The premium of self-reflection.* The best learning experiences occur when individuals are encouraged to explore their own thinking process and their intellectual and emotional relationship to knowledge, as well as the personal history that affects the learning experience. For example, in Project Inter-Actions, participants display and explain their projects, regardless of the stage that it is in. At the conclusion of each day of the workshop, participants were gathered into a *technology circle* where projects, thoughts, and learning processes were shared. In addition, there was a final open house where participants could share their projects with each other, but also with other family members and members of the community.

### ***Vygotsky's Zone of Proximal Development***

The concept of the zone of proximal development (ZPD) has been extremely influential in both child development research and educational research. Vygotsky (1978) defines the ZPD as “the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or with more capable peers” (p. 86).

The idea of the ZPD was especially influential in Project Inter-Actions, as each participant brought various skills to the workshop, even though the

technology itself was new to most. Some participants were stronger builders, others were stronger programmers, and still others were stronger designers. These strengths were not dependent on age and, for example, we were able to find children who were strong builders and parents who were weak programmers. Thus who was the more-capable peer varied depending on the task, and not the participant's age or developmental abilities. It was interesting to witness the differences in learning between children who worked with other children (and were only assisted occasionally by an adult) and children who worked constantly with their parents. These interactions between parents and children, and the differences in projects that the children created on their own versus the projects children created with their parents will be discussed in the results section.

### *Peer Learning Environments*

Peer learning environments, which occur when small groups of students learn from each other, can provide an environment rich in learning (Blumenfeld, Marx, Soloway, & Krajcik, 1996). In the specific case of Project Inter-Actions, we considered peers to include all of the participants, quite contrary to the normally understood definition of the word peer, referring to peers of the same age. Because most of the participants came to the first day of the workshop with the same level of knowledge about Lego robotics, regardless of whether adult or a child, we have a unique opportunity to observe learning that occurs between peers of various ages – we watched children interacting with other children, adults interacting with other adults, children interacting with adults, and both children and adults interacting with the more knowledgeable instructors. This is quite contrary to a traditional school setting where ZPD is normally considered; in a school setting, the teacher typically knows more than the children, who, in turn, are expected to learn from him or her.

Thus, constructionism, the zone of proximal development, and theories of peer learning provided the theoretical background for analyzing the experience of parents and children learning together about robotics technology while engaged in creating a personally meaningful interactive project during Project Inter-Actions' workshops.

### *Pilot Project*

In the spring of 2003, a first Project Inter-Actions pilot study was held at Tufts University's two early childhood centers – a full-day educational day-care center and a laboratory school for kindergarten, preschool, and first and second grades (Bers et al., 2004). The ten families of four- and five-year-olds recruited from these two centers represented a variety of religious backgrounds (including Jewish, Christian, Muslim) and cultural backgrounds (including Indian American, European, Chinese, and Caucasian).

The study was designed as a series of five free weekend workshops, each lasting approximately two and a half hours. In addition, at the conclusion of the project there was a weekday evening final open house so that the families could share and present their projects to other family members and to members of the community and the university (such as teachers, program directors, and staff).

Based on the four pillars of constructionism identified by Bers et al. (2002), Project Inter-Actions workshops were designed to allow participants to learn by hands-on, fun activities with concrete materials, to explore powerful ideas, and to allow participants opportunities for self-reflection upon their projects. Thus, each session of the workshop was carefully planned, using a project-based immersive methodology. After learning the essential skills to create a robotic project, the families were asked to create a final project that reflected their family's cultural background. Final projects included a water scooper, an Easter Bunny, and "Go-Lem, the Matzoh Robot."<sup>1</sup>

This pilot study gave us insights into how to design another study in which to explore how children and parents approach learning together. An important lesson from the pilot experience helped us craft the study that we report in this article: four and five year olds were capable of learning and working with the technology and their parents, however, they were a little too young to be able to work independently on the type of projects we were expecting them to create. In addition, since we wanted to conduct a controlled study, we needed to have two groups: one composed of parent/children dyads and another of children only. Therefore, we decided to recruit first- and second-graders who were slightly older than the children in the pilot study.

## METHODOLOGY

Based on the experience during the pilot project, a study was designed and implemented. In this study we recruited more participants and we had a more focused research methodology than in the pilot study, even though in educational research it is not possible to truly isolate variables in order to conduct a controlled study. We also focused on three different areas: (a) the differences in learning between children-only workshops and parent-child workshops, (b) the influence of a theme-constraint on the creation of a final project, and (c) the differences in learning in a weekend workshop versus a during-school setting. In this article, we will be focusing on the first area – the differences in learning between the child-only workshops and the parent-child workshops. We will first discuss how the workshops were organized and how participants were recruited and then we will discuss how the data was analyzed.

### *Description of General Workshop Organization and Method of Recruitment*

In the early spring of 2004, both parent-child dyads and individual chil-

dren (first- and second-graders) were recruited from the two campus schools, as well as at local elementary schools. Five two-hour workshops were held each weekend for five weekends, with the last hour of the fifth workshop being an open house. On both Saturday and Sunday there was one workshop for children-only (the morning session), and one workshop for parents and children (the afternoon session). In addition, the Saturday workshops were technology-only (the final projects could be whatever the participants wanted), while the Sunday workshops had an additional cultural component (the final projects were supposed to reflect an aspect of the child's or family's culture) (see Table 1).

In an attempt to have randomization of the families for research purposes, the families did not know that there was a cultural component to the Sunday workshops when registering (i.e., they did not know there was a theme constraint on the final project). There was a small registration fee, to cover the cost of paying the instructors and the materials. Before the start of the workshops, each family (including the families of the children in the child-only workshops) was sent a package which included a letter describing the workshop and outlining the expectations for both the parents and the children, a research consent form with a stamped envelope to be returned to the project coordinator, pre-questionnaires for both the parents and the children (to be handed in on the first day of the workshop), and explanations and guidelines for the Lego kits. Participants were able to borrow the Lego kits and take them home with them to keep working on their projects at home.

The workshops were also taught at the laboratory school on campus. Instead of a five-session weekend workshop of two hours, the sessions at the school were an hour long, twice a week, for eight weeks. The curriculum was relatively similar, though changes had to be made to adapt it to a shortened classroom session (Staszowski & Bers, 2005). Because of the differences in environment and teaching between the laboratory school sessions and the weekend sessions, this article will only focus on latter.

**Table 1**  
Summary of Project Inter-Actions Workshops

Workshop Day		Morning session	Afternoon session
Saturday	Participants	Children only	Parents & children
	Theme	Technology only	Technology only
Sunday	Participants	Children only	Parents & children
	Theme	Technology & culture	Technology & culture



### *Description of Participants*

In all, there were 76 participants at the start of the study: 20 parent-child dyads, 20 weekend children, and 16 children from the laboratory school. Similar to the pilot study, the participants came from a broad range of religious backgrounds (including Catholic, Muslim, Atheist, Jewish, Jehovah's Witness, Hindu, Protestant, and Russian Orthodox) and cultural backgrounds (including Asian American, African American, Latino, Italian, Scottish, Ukrainian, and Caucasian).

For this study we did not intend to do analysis on gender differences. However, as we began our data analysis, we were struck by the differences in the gender make-up between the child-only and parent-child groups. Of the weekend participants, 70 percent of the children in the parent-child workshops were boys, and 85 percent of the participating adults were males, while 55 percent of the children in the child-only workshop were boys and 40 percent of the contact parents (i.e., the parents who were the primary contact for the research project) for the child-only group were male. More specifically, of the 20 parent-child dyads, 11 were father-son, 6 father-daughter, 3 mother-son, and none were mother-daughter. However, the adult participants did not all come from an engineering or technical background as one may first expect; they came from a wide variety of occupations including a doctor, a musician, an industrial designer, a biochemist, an artist, an entertainment producer, and an exhibit director.

While at this point we can only offer our intuitions as to why there is this disparity in the genders of the participating parents, we have two preliminary ideas. First, males drawn to this type of workshop were probably those who grew up in a household where they were encouraged to "tinker," to explore with hands-on materials, and thus they have more experience and are more comfortable with such materials as Lego bricks. This sentiment is echoed by McIlwee and Robinson (1992) when they wrote "Like so many other boys, their socialization had engendered in them [male engineers] a love of tinkering" (p. 26). Second, families, with their already busy schedules, are looking for opportunities in which fathers can spend quality time with their children (this was expressed by some of the participants in response to the question of what was their favorite part of the workshop: "Working with my son designing and building.") Given this anecdotal data, in the future, we intend to explore gender differences in parental and child involvement in technological workshops such as Project Inter-Actions .

### *Goals of Workshops*

To guide the workshops, there were four goals that the instructors shared with the participants: (1) to have fun working together, (2) to learn about new technology, (3) to learn how to build and program a robot, and (4) to find a meaningful final project to create (participants in the culture theme –

the Sunday workshops – were asked to choose a theme related to their own heritage). These goals were shared with the participants at the start of the first day, and the instructors used them to guide their teaching.

### ***Description of Location and Set-Up of Workshop***

As in the pilot study, the workshops were held in the Curriculum Lab at the Eliot-Pearson Department of Child Development at Tufts University. The room is large and open, though it is divided into different sections with waist-high cabinets. In one section we placed several large tables so that two to four people could work at each table (i.e., we made sure there was plenty of room to spread out). In another section there were ten computers, so that each child (or parent-child team) could have their own computer to work at. In this section we also had a large projection screen and a teacher computer hooked up to a projector so that we could teach the programming to the participants while they were at their computers.

### **Description of Materials**

#### ***Lego Mindstorms Robotics Invention Kit***

The use of technology as an educational tool is becoming widespread – at a most general level, schools today usually have computers for students to use, and possibly digital cameras, video cameras, digital microscopes, and so forth. The use of manipulatives as a teaching aid is evident at the youngest ages – most early childhood settings have building bricks, Pattern Blocks, or Cuisenaire Rods in order to allow students to build, to design, to experiment, and to be creative (Brosterman, 1997). Digital manipulatives are now supplementing traditional hands-on materials, but they also afford students the opportunity to explore ideas and concepts beyond what traditional manipulatives can provide (Resnick, 1998).

The use of robotic manipulatives can serve as an ideal hands-on tool for children to learn about science, technology, engineering, and math (STEM) concepts. Children can explore traditional concepts such as gears, levers, joints, motors, and sensors while integrating art materials, everyday objects, and personal touches, all in a project that is personally meaningful. While using robotic materials, children can learn STEM concepts in a fun and creative way, which are traditionally taught through boring and uninspired means.

For our study we used a commercially available robotics kit, called Lego Mindstorms Robotic Invention Kits. Each kit contains a variety of pieces in different sizes and shapes. Many of the pieces are immediately familiar to people who have played with Lego pieces, including beams, bricks, and plates. However, in the robotics kit there are several additional pieces, including motors, light sensors, touch sensors, wires, axles, and gears. Perhaps the most interesting and unfamiliar piece in the kit is the RCX – a large yellow and gray brick (measuring 2.5” wide, 3.75” long, and 1.5” high) that-

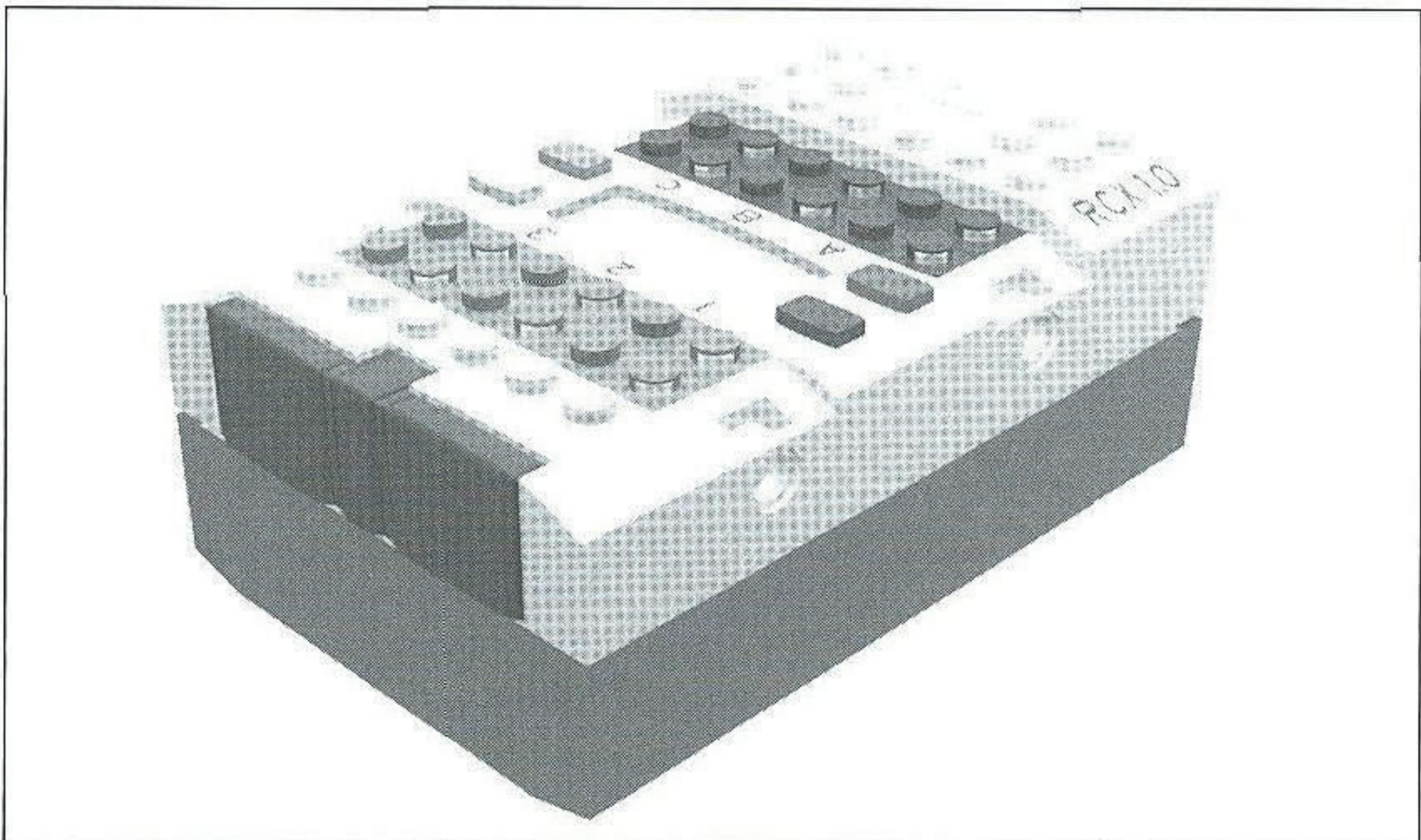
contains a micro-computer. The RCX has three input connections (for the touch and light sensors) and three output connections (for motors and lights). In addition, a LCD display provides information about the input and output connections as well as data that is stored in the processor. The RCX communicates information to and from a computer through infrared (the kit contains a USB infrared tower that connects to a computer) (see Figure 1).

When teaching about the function of the RCX, we analogized it as the *brain* of the project, and the wires that connect the RCX to the various inputs and outputs as the *nerves* (i.e., “Some nerves bring information to your brain, like how hot a pan is, and other nerves bring information from your brain, like ‘Release the hot pan!’”). This type of language helped the children understand the function of the RCX.

After signing a contract of responsibility to ensure that all the pieces were returned and giving a security deposit, each family was provided a kit to use for the duration of the project; they could take the kit home to use, but it had to be brought to each of the workshops and returned at the end of the last workshop. Participants were also given a variety of art materials and everyday objects to use in order to enhance their creations, such scrap fabrics, markers, construction paper, and yarn.

**ROBOLAB Software**

There are currently two main software packages that are used to program the RCX (and hence a robot); for Project Inter-Actions, we decided to use



**Figure 1.** Lego RCX: Inputs are the gray squares on the top, outputs are the black squares. The large black area in the front is the infrared port.

ROBOLAB™, a software program developed in partnership with the the Tufts University Center for Engineering Educational Outreach (CEEEO), National Instruments, and Lego Education, because most of the instructors were extremely proficient in its use. ROBOLAB is a drag-and-drop graphical interface that has several levels of difficulty, so the user can tailor the functions that are available to their personal skills (Portsmore, 1999) (see Figure 2). For *Project Inter-Actions*, even though we were only working with first and second graders, we decided to use the highest level of programming available, as we wanted participants to have access to the complete range of available functions. Several participants took a copy of the software to install on their personal computers so they could work at home.

### *The Instructors*

Each of the sessions had one graduate student as the lead instructor and three students as additional instructors. The lead instructors had participated in previous *Project Inter-Actions* workshops and had extensive knowledge of the materials. However, all of the instructors had knowledge about education and technology and had participated in a *Project Inter-Actions* training workshop. While some were undergraduate child development majors or graduate students in the teacher education program at Tufts, others were engineering students focusing on K–12 engineering education at the CEEEO. Thus in each session, there were instructors who were strong in the technology as well as instructors who were strong in teaching.

### *Overview of Curriculum*

The first two days of the workshops were dedicated to teaching the basic skills of building and programming, the third and fourth days were for building the final project, and the fifth day was an open house for sharing the final projects. During the first two days, the instructors taught skills to the group and then the participants had a mini-project to work on in order to reinforce the skills. For example, on the first day the participants learned how to build a car and program it to go forward and backward. Their activity was to create a car that could push “pizza toppings” (cut-outs of pizza toppings taped to Lego pieces) onto a “pizza” (a circle made on the floor with masking tape). At the end of each session, there was a technology circle – everyone



**Figure 2.** A simple program in ROBOLAB. This program would tell a car built with two motors to go forward for 4 seconds, then to go backwards for 2 seconds, then to stop.

would stop what they were doing (we often had to beg people to stop building and programming!) and join together in a circle. Under the direction of an instructor, each participant would share what they had built so far (it was stressed that the projects did not need to be completely finished, or even working to share), what they had learned, or any frustrations they came across. During the third and fourth days, the participants focused solely on their final projects – the instructors were available to help anyone who requested (or looked like they could use a helping hand).

Throughout all the workshops, and especially during the final two sessions, instructors encouraged participants to interact with each other for support, ideas, and help. For example, if a participant was struggling with a concept, and another participant had just solved the problem, the instructor suggested that they talk rather than have the instructor help. In order to support a strong peer-learning environment, the instructors kept the atmosphere non-competitive and often asked participants with new or innovative ideas to share them with the group. However, sometimes only an instructor could help, and in that case the instructors, keeping in mind the importance of the ZPD for both parents and children, tailored their teaching to a level that could be understood by the participant – neither too low, which would have made the participants feel inadequate for not understanding something easy, nor too high, which would have made the participants feel overwhelmed and incompetent. Thus, throughout the workshops, the instructors constantly tied their actions to the three guiding educational principles: the framework of constructionism, the concept of the zone of proximal development, and the concept of peer learning environments.

### *Examples of Final Projects*

On the last day of the workshop, each child or child-adult pair was asked to document their final project using a template in Microsoft Word that was created by one of the instructors. The documentation included a description of their projects (e.g., What does it do?), who helped them with their project, what they learned when making their project, and what was hard about their projects. Engaging participants in documenting their own projects was another venue to foster self-reflection. Digital pictures of the projects were added to the descriptions and posted on the project's website.<sup>2</sup> Child-only projects included a whale ("Speeder"), a music box, a rollercoaster, a soccer-ball kicker, a dragster, a windsurfer, a "digger," a windmill, a house, a giraffe, a nutcracker, an amusement park, an "egg beater ferris wheel," a "basketball crazy field," a hockey fan, a car, a Menorah, and a "moving garden". Parent-child projects included a "Mess Machine," a castle, "Uncle Feather" (a bird), a "Toyagumon" (Dijimon), a "basketball shooting robot," "Birdie," a robotic man, a catapulting car, a paddleboat, a "Vit Vit" (a singing and dancing robot that lives in a forest), "Art Mobile and Gear Mobile," "The Picker Upper,"

“Miss Violinist,” “The Frog Eater,” “Dancing Turkey,” “The Creature,” “Música: The Drummer,” and “The Christmas Tree”.

We provide two example projects below (note: these descriptions are copied exactly as written by the participants; no grammar or spelling changes have been made):

*Example 1: Ruwan's "Moving Garden" (Child-Only Culture Group)*

My project name is The Moving Garden.

My project just moves forward and backward.

It's important to me because I like to plant a lot.

It's also important because it brings happiness.

I first thought about how I could make a garden and how I could attach the garden to the car.

I decided to connect the garden at the back of the car.

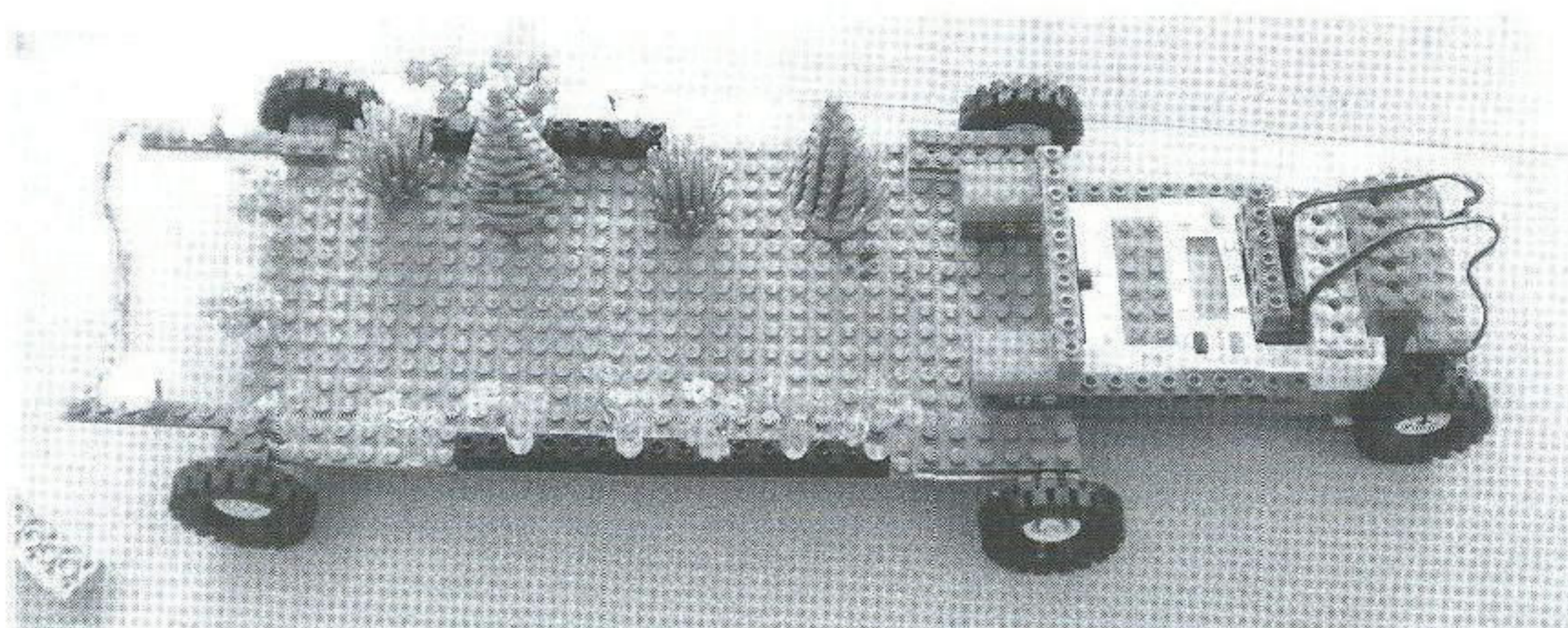
I did most of the programming at the Curriculum Lab.

I needed a lot of help with programming from my mom and the people at the workshop.

The hardest thing was keeping the car together.

I learned that you should never get frustrated and when something goes wrong always try again.

(See Figure 3)



**Figure 3.** Picture of Ruwan's "Moving Garden" (child-only culture group)

*Example 2: "Basketball Shooting Robot" by Edward & his Dad  
(Parent-Child Technology Group)*

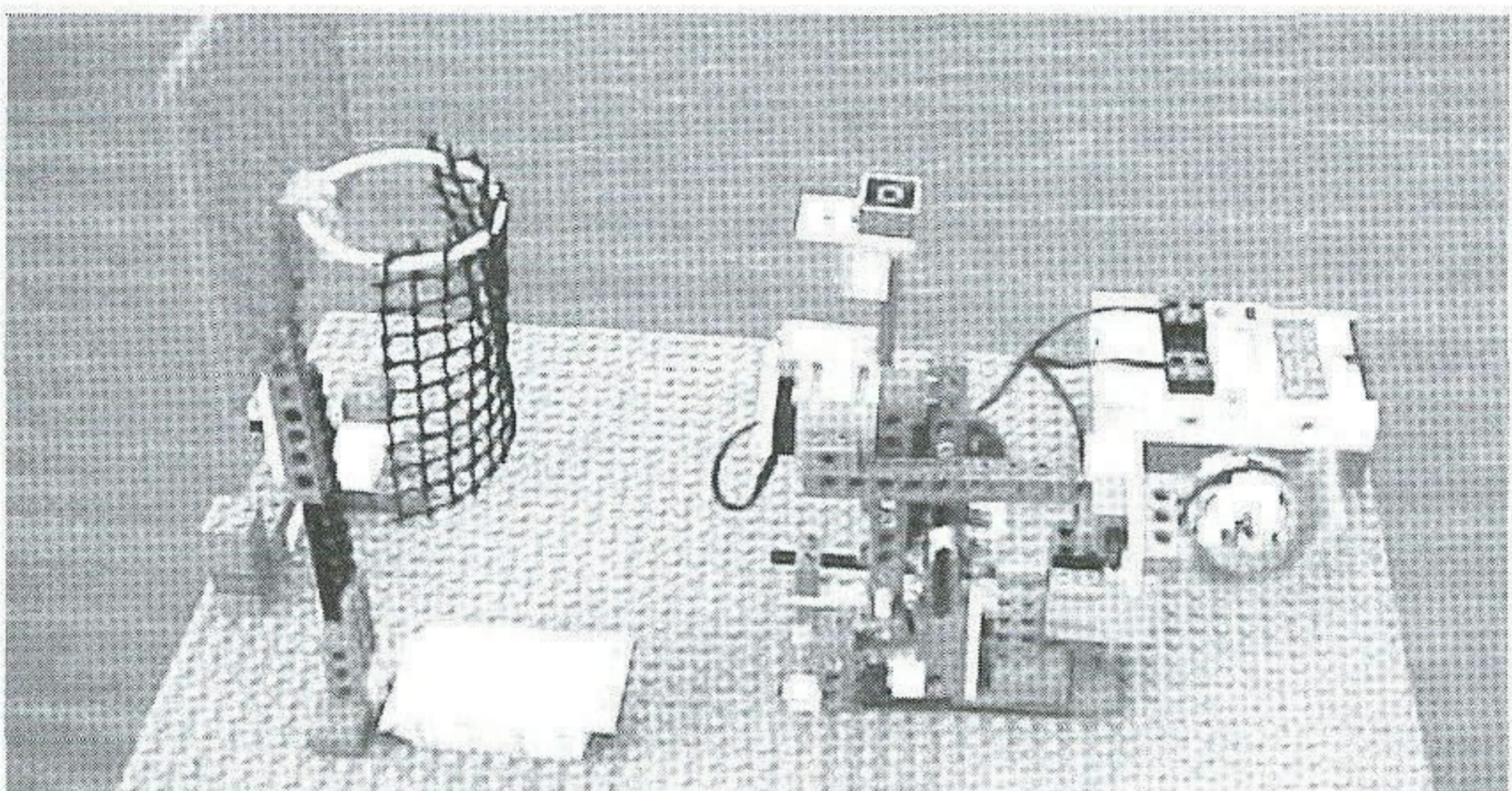
Our project is a robot basketball player that jumps up into the air (in slow motion) and shoots a basketball into the basket. It then lands on the ground ready to shoot another ball. Our project is important because it is cool. Edward started playing basketball this year, which was his inspiration for this project.

We built most of the robot in the second to last class, and we programmed it at home. We used a gearbox to make the robot go up and down. This is why it moves in slow motion. We used a second motor for the arm of the robot. We used timers to decide when to start and stop the motors. The program is pretty simple. We move the robot up for 3.3 seconds. Then we move the arm forward for 0.15 seconds. Then we move the arm backwards for 0.15 seconds and then move the player back down for 3.1 seconds.

The toughest challenge was to figure out how to make the basketball player move up and down. We first tried using teeth on the back of the player and a gear to move it up and down. We couldn't figure out how to attach the player to the ground and still have him move, so we went to plan B. We used a gearbox to turn a big gear that was attached to the player. This made the player move up and down. Another challenge was getting the timing right. We used TRIAL and ERROR to get the times right. It took many trials to get the robot to work without destroying itself.

We learned that it is possible to get a robot to throw a ball through the air accurately. We learned to keep trying if something doesn't work right the first (or the tenth) time.

(See Figure 4)



**Figure 4.** Picture of Edward and his Dad's "Basketball Shooting Robot" (parent-child technology group)

These two examples show two completely different levels of ability by participants – this will be discussed further in a later section.

## RESULTS

### **Collaborative Learning Between Parents and Children**

#### *Learning by Playing Versus Learning by Being Taught*

Throughout the course of the workshops, we observed many differences in building and programming with the robotic Lego bricks between the children and the adults. These differences will be explained in the following sections framed by the theoretical background presented previously. However, before presenting the data, we find it useful to share observational data collected by all the instructors regarding the two approaches taken by parents and children.

On the first day of the workshops, each of the participants received a new Lego kit; inside the new kit the Lego pieces come packaged in several plastic bags. The first task of the workshop was to open the kits, take out the bags, and explore the pieces while sorting them into their proper location using the diagrams provided in the kits. The children in the children-only sessions needed some help opening the bags, but they all began to investigate each of the pieces, many of which they had not seen before. Many of them made up uses for some of the pieces that were new to them, often trying to attach pieces where they traditionally did not attach. Bags of pieces were dumped on the floor, and the children were content just playing and making Lego structures without any instruction. We had to make them, for the sake of time, roughly sort their pieces so we could teach them the proper names of the pieces.

During the parent-child group however, the situation was very different. While the parents had an easier time opening the plastic bags, they sorted the pieces immediately (the children helped at times), and usually exactly as specified in the diagram. As soon as they were done sorting, they were impatient to get to the next part of the workshop so they could be told the uses and names of the pieces, even though their children wanted to play with the pieces.

Overall, this is the sense that we got from the workshop – the parents were always eager to get to the next part of the workshop, to be told what the lessons were so they could learn the more advanced skills. The children-only participants learned by playing and exploring the new materials, while parents needed direction and instruction. We should also note that over the course of the five weeks, three parent-child dyads in the parent-child groups left the project while no child quit the children-only workshops. The following answers given by two fathers on the post-questionnaire are indicative of many of the parents' beliefs:



*Question: Would you participate in a workshop like this again?*

Perhaps. It was a lot of fun but it was a big time commitment taking away from my child's free play time.

No – I like more guidelines and direction

Some of our colleagues at Tufts working with teachers and Lego Robotics have found the same situations – most teachers (adults) want to learn by being taught, instead of by playing and having fun with the materials (Cejka, 2005). Based on this anecdotal data, perhaps the question for future research study should be, “Have adults forgotten how to play?” In the following section we will provide data and describe how we analyzed results from this study.

### ***Methodology of Evaluation***

In order to analyze Project Inter-Actions, we used four main data sources: (1) videotapes of the workshops, (2) instructor memos that were written by every instructor at the end of each session, (3) photographs and descriptions of the final projects written by the participants and posted on the project website, and (4) pre- and post-questionnaires that were completed by all participants (and the parents of participants in the children-only sessions). In this article we report on the differences in projects between the parent-child and child-only groups and how this relates to learning.

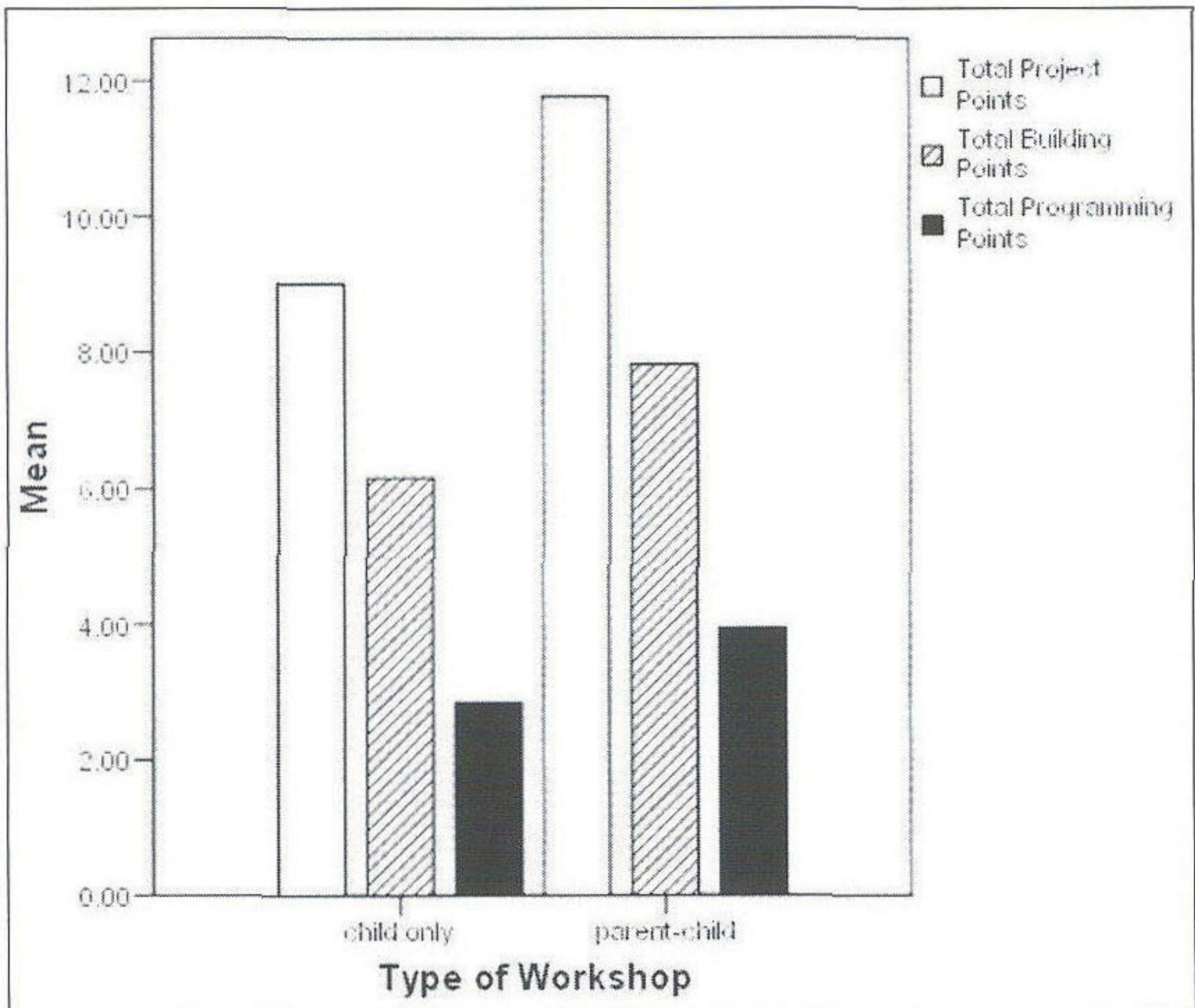
In addition to the original goals of Project Inter-Actions as mentioned above, we wanted to develop a method to evaluate robotic technologies for this and future projects. To do so, we created a scoring system that broke down building and programming, the two main components of the final projects, into the smallest elements that were important for each category. These elements were developed in collaboration with experts in the technology at the Center for Engineering Educational Outreach at Tufts University (Bers & Portsmore, 2005; Rogers & Portsmore, 2004; Portsmore, Rogers, Lau, & Danahy, 2004). The break-down of building included *gears, outputs, inputs, types of motion, success, and stability*, while programming included *outputs, wait-fors, structures, music, RCX communication, success, and containers*. Once the components for building and programming had been agreed upon, further collaboration led to developing a scoring system for each element. For each element, zero to two points could be given by the evaluator – for example, no points were given if no gears were used, one point if one type of gear was used, and two points if two types of gears were used. Summing up the points for building and programming resulted in three scores: building total points, programming total points, and total project points (the sum of building and programming total points). The complete scoring measurement can be found in Appendix A.

Projects were scored by watching the videos of the final project presentations and from the descriptions and pictures on the project webpage, as

well as by asking the instructors of the workshops when in doubt (i.e., if the building mechanisms were covered by art materials). Inter-coder reliability was established by having two raters overlap 10% of their rating samples (Hodson, 1999), which resulted in a substantial Cohen's kappa equal to 0.712 (Gardner, 1995).

***Influences of the Zone of Proximal Development and Peer Learning Environments When Parents and Children are Both Learning Something New***

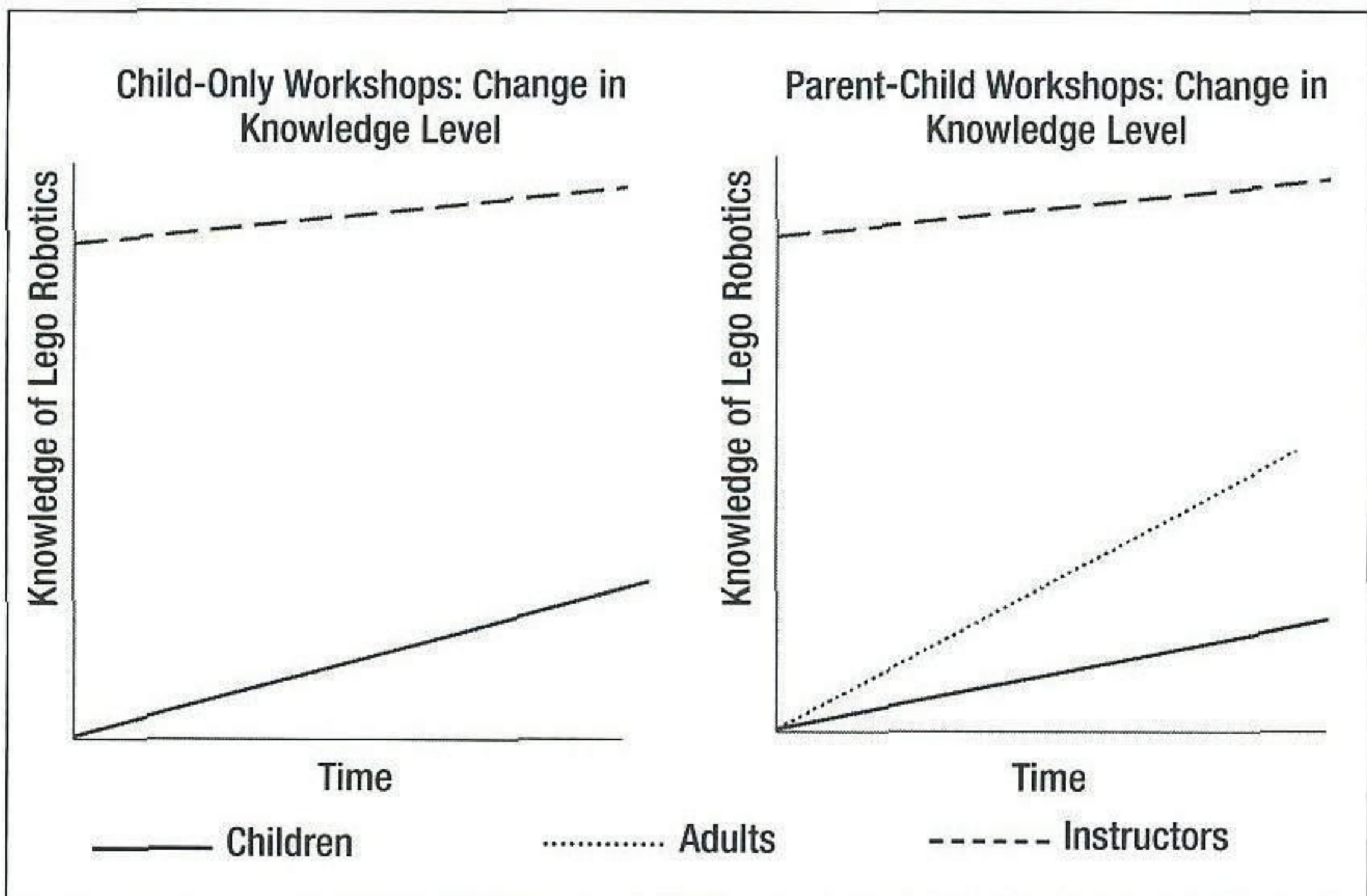
Using the above scoring mechanism, we found that there was a significant difference between the total project points for child-only and parent-child projects [ $t(35) = -3.462, p = 0.001$  (two-tailed)], total programming project points [ $t(35) = -2.456, p = 0.019$  (two-tailed)], and total building points [ $t(35) = -3.613, p = 0.001$  (two-tailed)]. Projects in the parent-child workshops scored significantly higher in the complexity of their projects (both in terms of programming and building) than projects in the children only workshops (see Figure 5).



**Figure 5.** Total project points, total building points, and total programming points for child-only and parent-child workshops

Based on these results and the theoretical framework set by the zone of proximal development, the first explanation would be that working with a more skilled peer allowed for more complicated building and programming to occur. However, we further suggest that in this project, parents actually prevented their children from learning. In a typical classroom, the teacher provides students with information that helps them achieve more advanced skills. In the PI workshops, however, there was an interesting dynamic: at the beginning of the project, there were two levels of knowledge, the participants and the instructors. In the child-only group, this dynamic continued throughout the entire project – when the children needed help with a building or programming skill, they typically asked one of the instructors for help. In the parent-child group, the initial two levels of knowledge turned into three, with the children, parents, and instructors each representing one level (since the parents understood more and faster than the children, and the instructors understood more than the adults). The following graphs attempt to visually show the change in level of knowledge of the participants in the PI workshops. Although parents had been exposed to the use of computers and basic Lego bricks for more time than the children (simply by being older), neither parents nor children had previous experience with Lego robotics (see Figure 6).

We observed that the total knowledge learned by children in the parent-child workshops was actually less than what the children learned in the child-



**Figure 6.** Comparison of the change in knowledge level in the child-only workshops and parent-child workshops

only workshops, even though the projects in the parent-child workshops were more complex than those in the child-only workshops. This finding is supported by an informal experiment that we conducted with the two groups during the third week of the workshops to evaluate the different levels of programming skills achieved at that point. At the beginning of the child-only session, we gathered the children and asked them to tell us what icons we would need to program a car to go forward for a certain length of time. We used printed large pictures of the basic icons used in ROBOLAB and we also had a large piece of white paper to which we could attach the icons. The children were all eager and enthusiastic about helping, and could easily tell us which icons to use and in what order. During the parent-child session, we replicated this experiment by also gathering all the children attending those workshops while an instructor talked with the parents about how the workshop was progressing. We asked the children the same question as in the previous group, but we were met with a lot of blank stares (see Figure 7).

This supported our observations from throughout the sessions: the children in the child-only groups understood how to do the simple things (like build and program a car) very well, while the children in the parent-child



**Figure 7.** Our informal experiment to observe levels of programming involved in making a simple car move

groups could talk to us about more complicated skills (such programming loops or music, concepts most children in the child-only sessions did not even know existed), but they could not complete the simple tasks as well as the child-only participants. For example, one boy in the child-only group was building a dragster for his final project; this conversation indicates the knowledge that the boy had learned about sensors:

*Instructor:* "Is that going to be your project or is that a practice project?"

*Child:* "My project."

*Instructor:* "What special things can your roadster do?"

*Child:* "Well, to finish it off we're going to put a black piece of tape, and then we're going to put light sensors in the front so that when it sees a black line it stops." (Pointing to his dragster)

*Instructor:* "Oh cool, that is a good idea. Very good."

*Child:* "And I'm going to use both light sensors."

*Instructor:* "Oh really?"

*Child:* "And put them on the sides so they can stop like that."

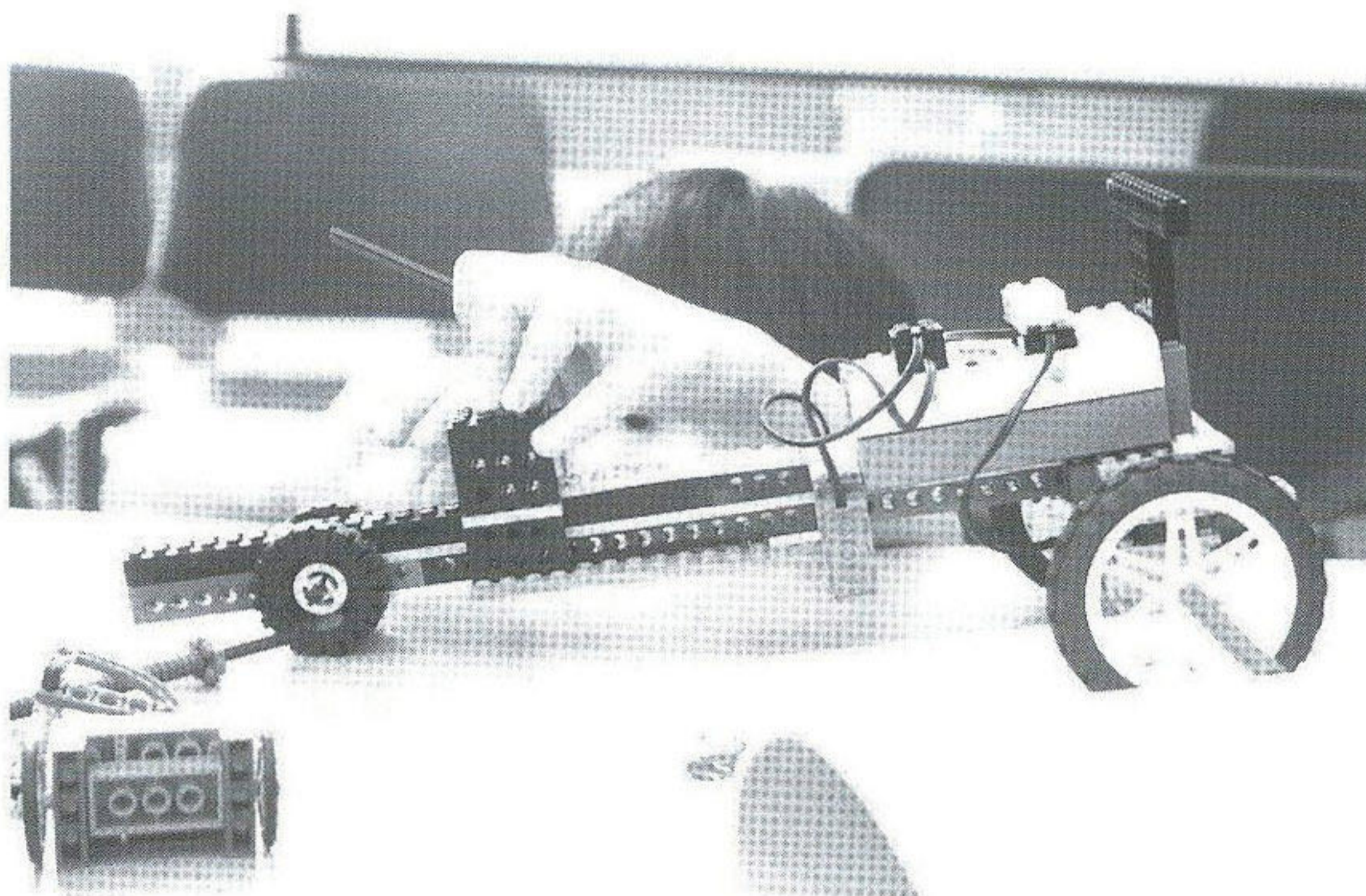
*Instructor:* "Cool. When you put the light sensor on do you think it is going to have to face out like before? Which way should it face?"

*Child:* "Down."

He, like many of the children in the children-only group, was able to articulate to us exactly what he wanted his project to do, but he was also able to build it himself (Figure 8).

We must consider, however, that the instructors were constant between the two groups, and thus, at least hypothetically, the teaching was the same for both groups. In fact, there was not a significant difference between the total project points between the Saturday and Sunday parent-child workshops,  $t(15) = 1.998$ ,  $p = 0.064$  (two-tailed). In addition, there was not a significant difference between the total project points between the Saturday and Sunday child-only workshops,  $t(18) = 0.184$ ,  $p = 0.856$  (two-tailed).

These findings illustrate an important aspect of the ZPD – if the more skilled peer provides help at a level that is too *high* above what the other peer (in this case the child) can understand and comprehend, the child will not learn as much (Berk & Winsler, 1995). This is an obvious statement for teachers who are very aware of the needs of scaffolding their teaching to meet their students at their own levels (Jacobs, 2001), however, it is not so obvious for parents with no pedagogical background in education or child development. In fact, many of the parents in the workshop (sixteen out of the original twenty) reported that



**Figure 8.** Boy in the child-only workshop building a “dragster.”

this was the first time they engaged in a workshop in which they were learning together with their children. The parents in the parent-child workshops had a difficult challenge: to find the fine line between understanding the concepts and pushing themselves harder and providing support and developmentally appropriate scaffolding for their children. We noticed that parents were so eager to learn (and maybe even eager to impress their children) that they forgot to guide or wait for their children along the way. We recall one parent who was absolutely determined to build a paddleboat, even though his son had no interest. The parent got so wrapped up in figuring out how to make a gear chain for his paddleboat, that his son decided to move to another part of the room to build his own Lego model – and the father did not notice! A mother, needing some help with their final project, tried to explain to us:

“So um, after we made the program to walk the dog, (*pauses and starts again*) the way this program works is as long as this touch sensor is free it will go forward and for however long you hold it, it will go backwards. That is what this top loop does. But ...after a while it [their project] becomes tedious because you have to chase after it to reach the green button to stop it. So what I’ve been trying to do is have a parallel task that would wait for ten seconds and then just turn off the motor, but nothing happened. *Continues to discuss how it does not do what she wants it do.*”

Notice how she used “I” even though her son stood by her and listened, fiddling with a Lego piece. This was a mother who throughout the workshop had paid particular attention to her son and had really tried to work with him every step of the way. As another example of the challenge parents faced when learning with their children, one parent wrote on the post-questionnaire that his least favorite part of the workshop was “The disappointing learning level post-day one.” Another parent’s response to the same questions perhaps best (and most humorously) sums up the situation: “The fact that the parents were on their own programming/design ego trips.” These parent’s responses match what we found while scoring projects and conducting our informal experiment: While the participants in the parent-child workshops created more complex projects, the children in these groups actually learned less than the children who worked on their own.

### CONCLUSION

Results presented in this article bring to light insights about how parents and children learn together in a technology-rich environments. However, this study had limitations such as the pre-selected nature of the participants (i.e., only adults interested in working with technology and their children participated) and the fact that the projects were analyzed after the workshops was completed. Do parents, when encountering a new technology, give precedence to their learning over their children’s learning? Do parents, who are eager to get to an advanced knowledge level, deprive their children of the opportunity to learn by taking over (i.e., are the children “pulled” out of their ZPD so that they cannot learn)? Can teachers, because they are not necessarily learning something new themselves, better judge children’s ZPD and thus provide a better learning experience for children?

We hope that findings from this study will shed light over different ways in which parents and children can spend time together learning something new – learning about a new technology is part of a larger picture of collaborative family learning. Children’s fascination with new technologies provide a unique opportunity to engage families in learning together. However, just as teachers go through years of school to learn how to teach, the more opportunities parents have to learn with their children, the better their collaboration will become.

### References

- Berk, L. E., & Winsler, A. (1995). *Scaffolding children's learning: Vygotsky and early childhood education*. Washington, DC: NAEYC.
- Bers, M., New, B., & Boudreau, L. (2004). Teaching and learning when no one is expert: Children and parents explore technology. *Journal of Early Childhood Research and Practice*, 6(2). Retrieved June 1, 2005, from <http://ecrp.uiuc.edu/v6n2/bers.html>.

- Bers, M. U., Ponte, I., Juelich, K., Viera, A., & Schenker, J. (2002). Teachers as designers: Integrating robotics into early childhood education. *Information Technology in Childhood Education Annual, 2002*(1), 123–145.
- Bers, M. U., & Portsmore, M. (2005). Teaching Partnerships: Early Childhood and Engineering Students Teaching Math and Science Through Robotics. *Journal of Science Education and Technology, 14*(1), 59–73.
- Bers, M. U., & Urrea, C. (2000). Technological prayers: Parents and children working with robotics and values. In A. Druin & J. Hendler (Eds.), *Robots for kids: Exploring new technologies for learning experiences* (194–217). NY: Morgan Kaufman.
- Blumenfeld, P. C., Marx, R. W., Soloway, E., & Krajcik, J. (1996). Learning with peers: From small group collaboration to collaborative communities. *Educational Researcher, 25*(8), 37–40.
- Brosterman, N. (1997). *Inventing Kindergarten*. New York: Harry N. Adams Inc.
- Cejka, E. (2005). *Inservice teachers' approaches to open-ended engineering design problems and the engineering design process*. Unpublished master's thesis, Tufts University, Medford, MA.
- Cuban, L. (2001). Making sense of unexpected outcomes. Cambridge, MA: Harvard University Press.
- Erwin, B., Cyr, M., & Rogers, C. (2000). LEGO engineer and ROBOLAB: Teaching engineering with LabVIEW from kindergarten to graduate school. *International Journal of Engineering Education, 16*(3). Retrieved 27 May 2005 from [http://www.ni.com/pdf/academic/us/journals/ijee\\_05.pdf](http://www.ni.com/pdf/academic/us/journals/ijee_05.pdf).
- Feldman, D. H. (2004). Piaget's stages: The unfinished symphony of cognitive development. *New Ideas in Psychology, 22*, 175–231.
- Furman, W., & Buhrmester, D. (1992). Age and sex differences in perceptions of networks of personal relationships. *Child Development, 63*(1), 103–115.
- Gardner, W. (1995). On the reliability of sequential data: Measurement, meaning, and correction. In John M. Gottman (Ed.), *The analysis of change*. Mahwah, NJ: Erlbaum.
- Hodson, R. (1999). Analyzing documentary accounts. *Quantitative Applications in the Social Sciences Series, 128*. Thousand Oaks, CA: Sage.
- Jacobs, G. M. (2001). Providing the scaffold: A model for early childhood/primary teacher preparation. *Early Childhood Education Journal, 29*(2), 125–130.
- McIlwee, J. S., & Robinson, J. G. (1992). *Women in engineering*. Albany, NY: State University of New York Press.
- Papert, S. (1980). Powerful ideas in mind-size bites. *Mindstorms: Children, computers, and powerful ideas*. NY: Basic Books, 135–155.
- Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. New York, NY: Basic Books.
- Papert, S. (1996). *The connected family: Bridging the digital generation gap*. Marietta, GA: Longstreet Press.
- Portsmore, M. (Spring/Summer 1999). ROBOLAB: Inuitive robotic programming software to support life long learning. *APPLE Learning Technology Review, 26*-39.
- Portsmore, M., Rogers, C., Lau, P., & Danahy, E. (2004). Remote sensing and tele-robotics for elementary and middle school via the Internet. *Computers in Education Journal, 14*(2), 72–75.
- Resnick, M. (1998). Technologies for Lifelong Kindergarten. *Educational Technology Research and Development, 46*(4).



Rogers, C., & Portsmore, M. (2004). Bringing engineering to elementary school. *Journal of STEM Education*, 5(3&4), 17–28.

Staszowski, K., & Bers, M. (2005). The effects of peer interactions on the development of technological fluency in an early-childhood, robotic learning environment. Proceedings of the American Society for Engineering Education Annual Conference and Exposition, American Society for Engineering Education.

U.S. Census Bureau. (2004–2005). *Statistical abstract of the United States, Section 24: Information and communications*. Retrieved May 26, 2005 from <http://www.census.gov/prod/2004pubs/04statab/infocomm.pdf>, p. 732.

Vygotsky, L. M. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.

**Notes**

<sup>1</sup>For additional information regarding the pilot project, please visit [http://www.ase.tufts.edu/devtech/Project\\_InterActions/Family\\_Projects\\_Page.htm#pilot](http://www.ase.tufts.edu/devtech/Project_InterActions/Family_Projects_Page.htm#pilot).

<sup>2</sup>[http://www.ase.tufts.edu/devtech/Project\\_Inter-Actions/Family\\_Projects\\_Page.htm](http://www.ase.tufts.edu/devtech/Project_Inter-Actions/Family_Projects_Page.htm)

**Appendix A.**

*Building and Programming Scales Used to Score the Final Projects.*

**BUILDING**

	Gears	Outputs (Motors, Lights)	Inputs (sensors)	Different Types of Motion	Success	Stability
<b>0 Points</b>	No gears used	No motors or lights used	No sensors used	Project does not move	Project does not work, or only works occasionally	Project does not stay together
<b>1 Point</b>	1 type of gear was used (i.e. may use several gears, but all for the same purpose)	One type of output used (i.e. only motors, or only lights)	One type of sensor used (i.e. only touch, only light)	Project has one type of motion (i.e. drives forward or backward)	Project works half of the time	Project falls apart half of the time
<b>2 Points</b>	More than 2 types of gears were used	More than one type of output used (i.e. lights and motors)	More than one type of sensor used (i.e. touch and light)	Project has more than one type of motion (i.e. goes backwards & has an arm that moves)	Project works most of the time	Project does not fall apart

## PROGRAMMING (INVENTOR 4)

	Outputs	Wait-fors	Structures	Music	RCX Communication	Success	Containers
<b>0 Points</b>	Program only controls motors or lights for a single power lever	Program does not use wait-fors	Program uses no structures	Program does not have music	Program does not use RCX communication	Program does not work as intended	Program does not use containers
<b>1 Point</b>	Program varies power levels, directions, function of outputs based on a stimulus. (On when light > 50, off light <50)	Program uses only one type of wait-for (i.e. only time, only light)	Program may use 1 simple type of structure (i.e. jump, task split)	Program includes some notes or a pre-programmed scroll	Program uses basic RCX communication (i.e. sending mail)	Program does half of what was intended	Program may use containers simply (i.e. only one use)
<b>2 Points</b>	Program uses sensor feedback to determine speed, power level (speed is proportional to light sensor value. <20, motor off, 20-40, speed 1, 40-50, speed 2, etc)	Program uses several types of wait-fors (i.e. time and light)	Program uses many simple structures, or uses 1 or more types of complex structures (i.e. events, forks)	Program includes a complete programmed song	Program uses complex RCX communication (uses feedback from another RCX in its program)	Program does all of what was intended	Program uses complex containers (i.e. container containers, information controls other pieces of the program)