

## 触手可及机器人技术方案： 为幼儿设计的应用计算机思维的方案

Marina U. Bers  
Tufts University

### 摘要

本篇论文描述了为幼儿设计的触手可及机器人技术方案。在十多年研究的基础上，本方案信奉这样的信念，即教儿童人造的世界、技术和工程学领域的知识，与教他们自然界、数字和字母一样重要。触手可及机器人技术方案把机器人技术作为手段来吸引儿童参与有助于发展计算机思维和学习工程设计的过程。它包含了一个建立在研究基础上、与儿童发展相适宜的机器人工具箱，儿童可以用来制作机器人并设计他们的行为。课程已在小学的学前班、夏令营和实验室情境下进行过预研究。笔者陈述了为触手可及机器人技术方案的理论框架提供指导和作为课程基础的计算机科学和机器人技术方面的“重要概念”，并把这些重要概念与其它学科领域和儿童的发展特点联系起来。文章还有对课堂教学理念和活动，以及用作评估学习成果的评估工具的描述。

### 导言

技术在我们周围随处可见。但在小学低年级时，儿童对这门学科知识学的很少。尽管经常看到幼儿使用硬纸板或者回收材料搭建城堡和桥梁，并成为“小小工程师”（Bers, 2008b），过去几十年来早期儿童课程关注的是语言和数学能力，直至近来才开始关注科学。本文中描述的触手可及机器人技术方案信奉这样的信念，即教儿童人造世界与教他们自然界、数字和字母同样重要（Bers, 2008a）。

但是，什么是我们人造世界的独特之处呢？是电子学、软件与机械结构的融合——即机器人技术学科。

## 触手可及机器人技术方案的背景

### 机器人和机器人技术

机器人不再是科幻小说中的形象，在很多地方都可以发现它的踪影。他们有着不同的类型——从从事工业生产到拟人机器——自动操作或者按预设的任务做事，诸如放射性垃圾清理、进行外科手术和汽车生产。对于什么样的机器可以被看作是机器人，人们目前还没有一致的看法。但大家都认为，机器人会做下面所说的部分或全部的任务——四处移动、操作机械手、或对环境作出感应和反应、并展示出模仿人类或其他动物的智力的和/或社会性行为（Craig, 2005）。

机器人技术这一学科为幼儿提供了学习机械学、传感器、马达、编程和数字领域的机会。随着机器人技术的日益普及，教育机器人工具箱在高中、初中和小学学校的使用越加普及（Rogers, Wendell, & Foster, 2010）。在本文中，我描述了一个为幼儿打造的机器人工具箱，它包含两个要素——在物质世界的建构和为建构能够互动和对刺激作出反应的人工制品的行为编制程序。商业公司如乐高以及很多大学的研究实验室已开发了教育机器人工具箱（Rogers & Portsmore, 2004; Martin, Mikhak, Resnick, Silverman, & Berg, 2000; Rusk, Resnick, Berg, & Pezalla-Granlund, 2008）。

在塔夫茨大学的戴夫科技研究实验室，我们得到国家科学基金的资助，专注于开发早期教育中可以运用的具有发展适宜性的机器人工具箱（Horn, Crouser, & Bers, 预出版）。与此同时，我们开发了把机器人技术引入课堂的课程和教学法（Bers & Horn, 2010; Bers, 2008a），并研究课堂中运用机器人技术的学习成果。我们的方法是通过邀请幼儿建构自己的机器人项目，比如追随灯光的汽车、根据感应原理工作的电梯和能演奏音乐的木偶，让他们参与游戏性的学习（Bers, 2010）。通过探索齿轮、杠杆、马达、传感器和程序环，幼儿可以成为工程师；通过创造他们自己的、能对环境作出反应的物体，他们变成能讲故事的人（Bers, 2008b; Wang & Ching, 2003）。

机器人技术可以成为学习实用性数学概念、科学探究方法及问题解决的途径

(Rogers & Portsmore, 2004)。再者，机器人技术让幼儿在玩乐中学习和学习中玩乐的过程中，参与与社会的互动和协商 (Resnick, 2003)。教育机器人工具箱是建立在蒙台梭利和福禄贝尔（他们早期的“教具”和“恩物”是设计用来加深儿童对诸如数字、大小和形状概念的理解）的传统之上的新一代学习用具 (Brosterman, 1997)。今天，大多数的早期儿童环境里的库氏数棒、模式积木、数字积木和其它学具都是通过精心设计来帮助儿童建构和进行试验的。最近，“数字式操作学具”扩展了儿童能够探索的概念范围；研究者们已经把计算能力嵌入诸如积木、珠子和球这类玩具中，因此幼儿可以学到动态的过程和“系统化概念”，比如之前看来对儿童过于复杂的反馈信息和萌发的概念 (Resnick, Berg, & Eisenberg, 2000)。

在这样的传统中，机器人技术提供了一个把儿童引入技术和工程学世界的机会。机器人技术操作材料将儿童引入促进精细运动技能和手眼协调，以及蕴含了合作和团队精神的活动中。它们也提供了一个具体的触手可及的方式来理解抽象的概念 (Bers, 2008a)。例如，在把玩机械部件以设计他们的机器人生物时，儿童会探索杠杆、连接和马达，而且能制作简单的机器。通过把齿轮加到机器中，他们开始探索比率这一数学概念。

### 计算机编码、计算机思维和儿童

运用机器人技术所涉及的远远超过建构物理性质的人工制品。让机器人“活起来”涉及计算机程序。因此，儿童学着去制作计算机程序——使机器人移动、感受并对环境做出反应的算法或者指令的顺序。

对儿童计算机编程的研究始于几十年前的麻省理工学院人工智能实验室，该实验室后来成为乐高实验室，那时西莫·帕佩特 (Seymour Papert) 发明了一种小海龟，儿童可以通过使用文本乐高编程语言来控制小海龟 (Bers, 2008a)。近来研究表明，年龄低至四岁的儿童都可以理解基本的计算机编程概念，并能组建简单的机器人 (Bers, Ponte, Juelich, Viera, & Schenker, 2002; Cejka, Rogers, & Portsmore, 2006)。早期的乐高研究表明，当有系统地介绍计算机编程时，它可以帮助幼儿提高视觉记忆和基本的数感，也能发展解决问题能力和语言能力 (Clements, 1999)。帕佩特的研究(1980)和雷斯尼克 (Resnick, 1996) 的研究也表明，学习如何编程或可改变人们的思维方式。

计算机思维是一种分析性思维，它与数学思维（例如，解决问题）、工程学思维（设计并评估过程）和科学思维（系统化分析）有许多相似之处。这一术语源于帕佩特和他的同事对于以设计为基础的建构主义编程环境的开创性工作，它指的是运用算法来解决问题的方式，也指科技能力的获得（Papert, 1980, 1993）。计算机思维的基础是抽象化的——将案例中的概念抽象化并评估和选择“正确的”抽象概念。它有赖于对输入（对不同变量和计算机指令的操作）的选择，对输出（成果信息）的观察，以及对二者之间所发生的情况的分解。计算机思维包括从计算机指令（编程语言）提取到计算机行为的能力，发现导致错误的潜在的“漏洞”和错误的能力，以及决定在输入—计算—输出算法中哪个细节应该要突出，哪个要保持，哪个要丢弃的能力（Wing, 2006）。

在对建构主义编程环境进行研究的文献中，可以发现对于小学低年级儿童的计算机思维的早先的研究（Repenning, Webb, & Ioannidou, 2010; Resnick et al., 2009），温（2006）认为计算机思维是每个人都可以掌握，而不仅仅是一种计算机专家独有的基本技能：

为了读、写和算，我们应该在儿童的分析能力中增加计算机思维能力。正如印刷技术推动了三 R（译者注：指读、写、算）的推广，……计算和计算机促进了计算机思维的推广。计算机思维通过利用计算机科学的基本概念，牵涉到问题解决、设计系统和理解人类行为（Wing, 2006, p. 33）。

总括而言，计算机思维指的是一系列反映计算机科学领域广度的心理工具。

## 触手可及机器人技术方案

### 方案目标和研究问题

触手可及机器人技术方案是一个教育机器人技术方案，它已经在学前班到小学二年级的师生中进行过试验。它由课程、评估工具和有着发展适应性界面的机器人技术建构工具箱组成。课程和工具箱旨在向幼儿教授一组特定的心理工具——即在把计算机思维应用于机器人情境时的有用的重要概念和技能。

对于新技术是否可能促进幼儿的计算机思维，以及哪种学习轨迹能获得最好的成果，人们目前知道的并不多。这些正是触手可及机器人技术方案的目标，这

个方案既探索学习方面，又探索有关计算机界面设计的事宜，如触手可及的编程语言。

触手可及机器人技术方案研究的主要有三个核心问题：

- 当幼儿接触到一个机器人技术教育方案时，幼儿的计算机思维的发展轨迹是什么？
- 在机器人技术方案中包含了哪些幼儿能够发展的概念和技能，他们需要怎样的支持机制（从课程和技术两个角度考虑）？
- 为了让幼儿获得成功的学习经验，一个发展适宜性的机器人技术工具箱应该包括哪些设计元素？

本文并非直接应对这些研究问题；相反，我们旨在描述使我们有可能探索这些问题的触手可及教育方案。想获得有关这一方案的更多信息，课程、技术和研究论文等，可以参考以下网站 <http://ase.tufts.edu/DevTech/tangible/>。

### **积极的技术发展（PTD）：方案的理论基础**

积极的技术发展（PTD）框架构成了本方案的理论基础。PTD 是一种跨学科的方法，它把应用性发展科学和积极的儿童发展的研究与把计算机作为媒介的交流、计算机辅助的合作性学习和技术领域中的建构主义学习观念融合起来。PTD 是已经影响了教育技术的计算机通用知识和技术素养运动的自然延伸（Pearson & Young, 2002），它将心理和伦理元素加入到认知的维度中（Bers, 2008a; Bers, 2006; Bers, Doyle-Lynch, & Chau, in press）。PTD 考查了在数字时代中长大的儿童的发展性任务，并提供了一个模式去设计和评估有着技术特色的活动。它阐明了以技术为基础的教育方案和经验的设计和评估，这些方案和经验旨在帮助幼儿积极地使用技术，学习新东西，创造性地表达自我，与人沟通，关爱自己和他人，以及对集体有所贡献，与此同时，发展建立在个人价值观和道德观基础之上的自我认同感。建构主义理论和积极儿童发展理论这两方面的理论都对 PTD 的框架产生了影响。

建构主义。建构主义出自于 20 世纪 60 年代由西莫·帕佩特主持的麻省理工大学乐高小组进行的有关智力轨迹的研究（Bers et al., 2002）。建构主义有四大“支柱”。首先是对“在实践中学习”的教育方法。在很大程度上以皮亚杰的建构主义为基础，它强调通过使用新技术来帮助儿童在实践中学习，从积极探究中学习，

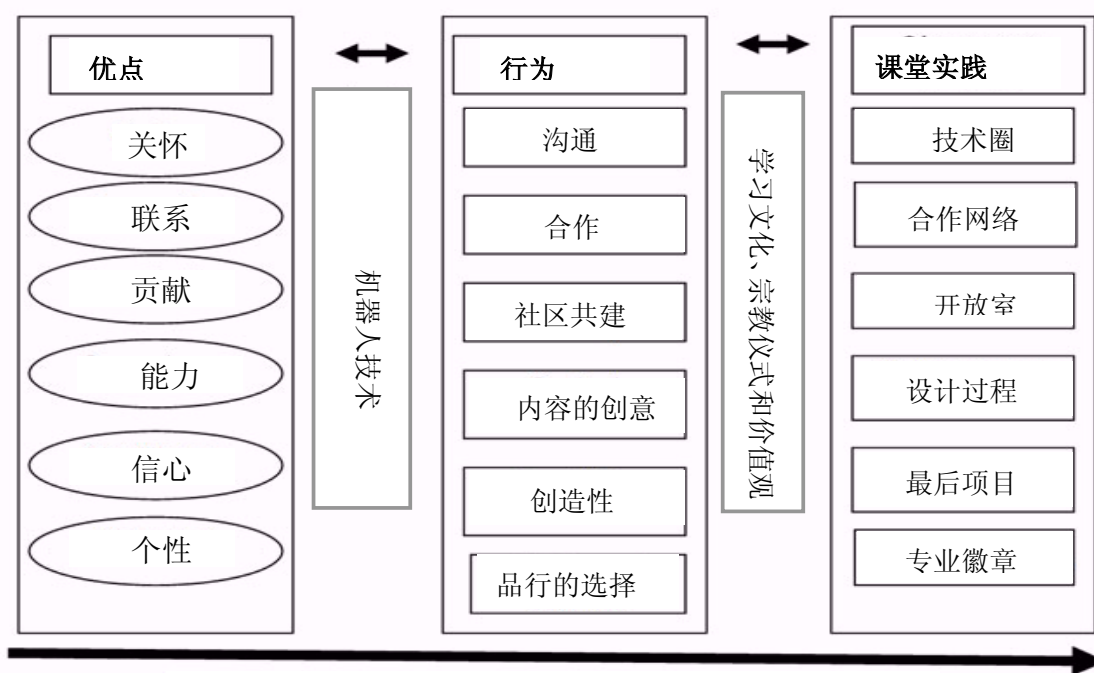
从游戏中学习。建构主义的第二根支柱是对物体对思维的重要性的认知，即物体能够支持思维的具体方式的发展以及对抽象现象的学习。计算机和机器人建构工具箱作为强有力的工具，在设计、创造和操作现实世界和虚拟世界里的物体有着突出的作用。建构主义的第三根支柱是核心概念能给个体的学习和理解赋权。“核心概念”是在认识论和个人层面上能立即产生影响的某个领域的中心概念，它与其它学科领域相互联系，并且植根于儿童在很长时间内内化的直觉知识（Papert, 2000; Bers et al., 2002）。建构主义者认为，计算机是新观念的一个强有力的载体，这些观念能提供新的思维方式、新的使用知识的方式、以及新的与其它领域的知识建立个人的和认识论上的联系（Papert, 2000）。建构主义的第四根支柱是对自我反省重要性的认识。从一个建构主义者的视角来看，计算机的编程是一种对思维如何运作，反思自己的思维过程以及如何对一个人的智力、情感与知识之间的关系获得新的洞察力的有效的方式（Papert, 1993; Kafai & Resnick, 1996）。

帕佩特的建构主义随着他的开创性著作《头脑风暴：儿童、计算机和核心概念》（Papert, 1980）的发表，在上世纪八十年代在教育界广为人知。在《头脑风暴》中，帕佩特提议把计算机编程作为儿童学习数学的途径，更重要的是，儿童可以把它作为理解自己的学习的一种方式。通过设计和修正计算机程序（即对外部物体进行反思）的过程，儿童可以发展的不仅是计算机思维，还有关于问题解决和学习的元认知的方法。

积极的儿童发展（PYD）。触手可及机器人技术方案的研究关注的是个体与环境间的动态关系，强调幼儿的长处和优点，而不是专注于减少或阻止他们的冒险行为（Damon, 2004; Larson, 2000; Theokas & Lerner, 2006; Scales, Benson, Leffert, & Blyth, 2000）。“积极”一词意味着对美好品性和活动（例如，发展上的优点）的提升，从而将幼儿引向一条好的发展轨迹中（例如，向着提升自我和社会性发展）。勒纳等人（Lerner, Almerigi, Theokas, & Lerner, 2005）把PYD模式的框架的发展性优点概括为“六个C”：能力(competence)、信心(confidence)、个性(character)、联系(connection)、关怀(caring)和贡献(contribution)，这些都是被人们视为营造一个繁荣且健康的社区的途径。

受到了建构主义理论和积极儿童发展理论的影响，积极技术发展理论（PTD）是一个多维的框架，它使“6个C”在我们的数字世界里凸现出来。它不仅强调

发展性的优点，也强调技术所支持的积极行为——内容的创意、创造性、沟通、合作、社区共建以及品行的选择（Bers et al., in press）。作为一个针对设计和实施教育方案的框架，PTD 考虑了协调教与学的学习环境和教育实践、文化价值观念和宗教仪式（Rogoff, 2003; Rogoff, Turkanis, & Bartlett, 2001）。下面将要描述图 1 所呈现的框架的元素。



在社会文化情境下个体发展轨迹

图 1: PTD 框架，包括优点、行为和课堂实践

### 触手可及方案和 PTD 框架

正如图 1 所示，触手可及机器人技术方案通过特定的课堂实践（右栏所示），即为幼儿提供机会参与特定的与技术有关的行为（中间栏）实践，强调“六个 C”，即积极儿童发展的优点（左栏所示）。

内容的创意。触手可及方案包括内容的创意：一位幼儿制作一个人工机器人并给它的行为进行编码。搭建的工程设计过程和涉及编程的计算机思维培养了幼儿在计算机素养和技术素养方面的能力。让幼儿在制作机器人的过程中坚持设计日志这一课堂实践有助于让幼儿（还有教师和家长）清楚他们自己的思维、学习轨迹、以及项目随着时间的演变进程。像科学方法一样，工程学设计过程的正式

步骤——提出问题、进行研究、实施计划、制作样品、进行测试、再次设计和分享策略——给学生提供有系统地应对问题的工具。触手可及设计日志可能通过支持这些步骤，提供具有不同程度的有计划的途径，从而帮助幼儿完成从理念到作品的过程。一篇日志可能包含强调所有设计过程的工作单或者有利于想象力的白纸；最理想的是能把二者结合起来，这种个性化的过程很重要。一些幼儿需要约束和自上而下的计划才能更有效地工作。一些幼儿并不喜欢事先计划。他们可能属于如“修补工”和“手工作坊者”，即与技术进行对话和协调的学习者（Turkle & Papert, 1992）；他们喜欢自下而上地工作，在他们创造、设计、搭建和编程时，他们借助胡乱摆弄材料来想出新点子。这两种认识论的风格对技术领域的建构能力都是有益的。

创造性。触手可及方法在解决问题上，是基于创造力的提升而不是效率的提升；这一方法可以用“工程学”这词的原始含义来阐释，“工程学”的含义源于拉丁词 *ingenium*，这个词的意思是，“先天品质、心理能力、聪明的发明”（Random House Webster’s Unabridged Dictionary, 2006）。方案将媒介如乐高块、马达、传感器、可回收材料、艺术品和手工艺材料，以及来自于编程语言的图表元素融合为一体。在运用这些媒介创造性地解决技术问题的过程中，儿童发展他们在学习潜能方面的自信心。然而，聪明的或创造性的项目是难于制作的，而且制作过程也会让人沮丧。多次尝试之后，儿童的机器人鳄鱼的嘴巴可能仍然打不开，或是她的汽车每次左转的时候都停下来。为了避免沮丧，有些教师谨慎选择儿童将要从事的方案，或者提供逐步的指导。这样的策略也许可以帮助儿童免遭艾伦·凯所说的创造性学习中的“苦之乐”（Kay, 2003）。与此相反，触手可及方案的方法旨在帮助儿童学习应对沮丧——此乃培养一个人对自我学习能力的信心的一个重要步骤。学习环境的创造旨在创造一种氛围，在这种氛围中，有些事情可能不一定成功，而且第一次就成功的现象纯属罕见，或许它表明了儿童对自己提出的挑战还不够。在儿童完成程序的过程中，他们逐步意识到他们有能力通过多次尝试，通过运用不同的策略，或者通过寻求帮助而找到解决方案（Bers & Horn, 2010）。

合作。大多数针对年龄较大的儿童的教育机器人方案，如国家机器人挑战和 FIRST（For Inspiration and Recognition of Science and Technology 对科学技术的灵



感和赏识), 都是为了竞赛而建立, 在竞赛中机器人必须完成指定的任务, 通常都以赢过其他机器人为目标。然而, 研究表明, 女性不擅长强调竞争的教学策略 (Turbak & Berg, 2002)。这样的策略在早期教育环境中也不一定是适宜的 (Bers, 2008a)。触手可及方案学习环境不是着重竞争, 而是促进儿童在团队中学习、分享资源并互相关怀。合作网的使用培养了合作精神。在每天学习开始的时候, 每个儿童在收到设计日志的同时, 一张个性化的页面也会随之出现, 页面的中间是他或她自己的照片, 教室里所有其他儿童的照片环绕着中间那张照片 (如图 2 所示)。在一整天里, 在老师的提醒下, 每个儿童从自己的照片出发, 在自己的照片和与他或她有合作的小朋友的照片之间连一条线 (在这里合作是指在一个项目中获得帮助或提供帮助、一起编码、借还材料或者在共同的任务中进行合作)。在一周结束时, 儿童会写或画一张“感谢卡”送给他合作最多的伙伴。



图 2. 一张空白的合作网的实例

沟通。沟通是触手可及方案的一个重要特征, 它包含了有助于提升同伴之间或者幼儿与成人之间的联系感的机制。鼓励沟通的一个特征就是技术圈。在技术圈中, 儿童和成人停下他们的工作, 把他们的项目成果放在桌上或者地板上, 围成一个圈坐下来, 然后交流他们项目的进展情况 (Bers, 2008a)。这与儿童在幼儿园里接触到的其他集体活动时间 (译者注: 在地上围坐成一个圈) 是相似的 (Kantor, Elgas, & Fernie, 1989)。技术圈提供了以团体的形式来解决问题的机会

——即为了解决编程或建构过程中的技术问题。教师通过要求儿童展示他们所做的项目，并问“有什么是预期达到，有什么不是？”，“你想要达到什么样的目的？”“为了达到目的你需要知道些什么？”之类的问题来组织技术圈活动。老师然后运用儿童的项目和问题来强调项目所表现出来的核心概念。课程则在这一特定学习共同体所需要的知识的基础上形成。这种根据需求提供所需的技术信息的方法，是区别于说教的一种方法。技术圈可以根据儿童的需要和教师引入新概念的需要，在项目开始起每 20 分钟或只在结束一天学习的时候进行一次。

社区共建。触手可及机器人技术方案中的社区共建技术有助于提供让每个幼儿对学习环境和社区做出贡献的网络。在瑞吉欧·艾米利亚方法（始于二战后意大利瑞吉欧·艾米利亚地区婴幼儿中心和幼儿园）的精神中，儿童的作品通过一间开放室、一个展览日或者展览的形式与社区分享。开放室给儿童提供了真实的机会与其他有助自己学习的人，如家人、朋友和社区成员，一起分享和庆祝他们的学习过程和可触及机器人技术的成果。这些公开的展示使其他人和儿童自己能够看到他们学到了什么。

品行的选择。触手可及机器人技术方案中的活动为儿童提供了机会试验一下“如果……会怎么样？”的情况，以及考虑他们的选择可能带来的结果。品行的选择不仅仅由儿童做出，老师也做出对儿童行为有重要影响的决定。例如，如果教师把乐高积木按种类分好，放在教室中央的大箱子里（而不是作为预先分好类的机器人工具箱分给每个儿童或者小组），儿童就能学会在不用找遍“最需要的”积木的箱子的情况下，拿到他们所需要的积木，如特殊的传感器，或者彩色的乐高小人物积木。他们也学着如何与他人协商得到自己需要的积木。对于推行触手可及机器人技术方案的老师来说，帮助儿童以一种公平而负责的方式发展一种内在的准则指导自己的行为与关注机器人技术的学习同样重要。方案强调品行的选择，可能激励对价值观的审视以及个性品质的探索。角色的区分对于一个有责任感的学习共同体的成长很重要。在任何班级，如一个儿童可能在机械学习方面非常快，而另一个儿童变成编程方面的专家，还有一个儿童呢，可能擅长解决问题或者很熟练地处理组员之间的争端。这些儿童很容易被教师或其他儿童贴上“专家徽章”的标签。这些被视作在某些方面特别擅长的儿童，能做出选择以帮助伙伴们构建一个更大的结构或者应对其它挑战。教师也鼓励儿童扮演新的角色和灵

活处事；有一个徽章叫做“尝试新事物的专家。”

### 对触手可及机器人技术方案课程的概述

触手可及机器人技术方案课程引入一种有计划的、发展适宜性的方式，并使用了机器人技术情境下计算机科学的六个核心概念——机器人技术、工程学设计过程、排序和控制流程、循环和参数、传感器，以及子程序。表1是对这些概念的描述。在这一课程的试验版中的主题——交通、社区、动物——是变化的，但计算机科学和机器人技术的核心概念不变。在此以有关交通的课程作为一个范例。

表1 核心概念、定义、活动、以及触手可及方案的学科联系			
核心概念	定义	活动	学科联系
机器人技术	关注能够接受指令并能完成任务的机器人或机器人的创作和编码的工程学学科。	<b>什么是机器人？</b> 通过观察不同机器人并讨论其功能对机器人进行介绍之后，儿童制作自己的机器人交通工具并探索部件与能用来编码的指令。	<ul style="list-style-type: none"> <li>• 工程学</li> <li>• 计算机科学</li> </ul>
工程学设计过程	一个用来开发产品以满足需求或解决问题的过程。它有几种可选择的步骤：明确需求或确定问题，进行研究，分析可能的解决方案，开发产品，沟通及呈现产品。	<b>实质性搭建：</b> 儿童搭建非机器人性质的交通工具，把小型玩具娃娃从家里送到学校。交通工具应是实质性的且能执行预设功能。 <b>设计日志：</b> 儿童使用设计日志来学习工程学设计过程。 所有编程活动。（如下所示）	<ul style="list-style-type: none"> <li>• 工程学</li> <li>• 计算机科学</li> </ul>
排序/流程控制	指令的顺序可以用一个程序来描述，且能通过机器人的行为表现出来。每块积木有着特定的意义。积木的顺序很重要。	<b>摇摇乐：</b> 学生选择适宜的指令并将其按顺序编入程序，以使机器人能随摇摇乐跳舞。	<ul style="list-style-type: none"> <li>• 创造性的故事讲述</li> <li>• 观念的组织</li> </ul>

			<ul style="list-style-type: none"> <li>• 数理证明</li> <li>• 程序性思维</li> </ul>
循环和参数	指令的顺序可以加以改变以使指令重复出现。控制流程指令以使新的信息合理化。例如，循环可以改变成无限制重复下去或重复特定的次数。	<i>多次重复直至我说当：</i> 学生使用一对循环积木（“重复”/“结束时重复”）以使机器不断运行，永不停歇，而且只在运行一定的数量时到达特定位置。	<ul style="list-style-type: none"> <li>• 自然界周期现象</li> <li>• 调度</li> <li>• 计时和控制</li> <li>• 反馈循环</li> <li>• 数感</li> </ul>
传感器	一个机器人可以使用传感器，就像人类的感觉器官，在它所处的环境中收集信息。感应信息可用于在机器人接受给定指令时的控制。	<i>通过隧道：</i> 儿童使用光传感器和指令来给机器人编码，使得在周围环境黑暗时打开灯，反之亦然。	<ul style="list-style-type: none"> <li>• 科学的观察</li> <li>• 因果联系</li> <li>• 传感器（既包括人造的也包括自然的）</li> </ul>
子程序	在一个程序的子程序中，一个机器人可以根据给定条件接受一组指令或者另一组指令。	<i>机器人决定：</i> 学生对机器人进行编码，以根据光感或者触感信息选择两个目的地中的一个。	<ul style="list-style-type: none"> <li>• 因果联系</li> <li>• 传感器（既包括人造的也包括自然的）</li> </ul>

触手可及课程包括最少 20 小时的课堂学习，根据上述的六个核心概念可以分为以下有计划的六个部分的学习内容：

- 第一部分 实质性搭建（工程学设计过程）
- 第二部分 什么是机器人？（机器人有特定的部件以执行指令）
- 第三部分 摇摇乐：指令的顺序（指令的秩序或者顺序）
- 第四部分。无限制重复直到我说当（循环和数的变量）
- 第五部分 通过隧道（传感器和循环）
- 第六部分 机器人决定（传感器和子程序）

表 2 描述了课程中的六节课,并对课程的学习目标用儿童能够理解和做出来  
的语言加以描述。

表 2 六节中每节课的目标		
课	学生将会理解……	学生将能够……
实质性搭建 (工程学设计过程)	乐高积木和其他材料可搭建坚固的建筑。 工程学设计过程对计划和指导人工制品的创造是有用的。	用乐高积木和其他材料搭建一个实质性的、非机器人的交通工具。 使用设计日志学习工程学设计过程。
什么是机器人? (机器人有特定的部件会执行指令)	机器人需要可移动的部件,比如传感器,以能够执行程序指定的行为。 机器人大脑 (RCX) 已经编好了指令以使机器人执行自己的行为。 机器人大脑必须为了实现传感器功能而与传感器进行沟通。	描述一个机器人的部件,包括机器人 大脑、传感器和电路。 通过可触及的积木或者图标,电脑界面,以及乐高 IR 塔,上传一个机器人的程序。 通过使用乐高积木和其他材料,搭建一个实质性的机器人的交通工具。
摇摇乐:指令的顺序 (指令的秩序或者顺序)	每个图标与特定的指令相符。 一个程序是一个机器人遵循的一系列的指令。 积木的顺序表明了机器人执行指令的顺序。	选择适合的积木以适应一个计划好了的机器人行为。 把一系列的积木连接起来。 在电脑中上传一个程序并把它传送给机器人。
重复运行直到我说当 (循环和次数的变量)	对指令或指令的顺序可以进行改变使它不断重复。 一些编程指令,如“重复”,可以修正以适应增加的新信息。 一个只用几块积木的简单程序比完成同样目标的一个更复杂的程序要好。	意识到需要使用循环程序的情景。 写一个含有循环的程序。 使用参数来修正该程序结束以前循环运行的次数。
通过隧道(传感器和循环)	通过使用传感器,一个机器人能够感受并看到它的四周。 一个机器人通过改变自己的行为对已收集到的信息作出反应。 植入机器人的程序可以使它保持执行一个固定的任务,在特殊的条件出现时才停止。	把一个光传感器或触感器联接到机器人 大脑的正确端口。 写一个含有等待某种条件出现的程序。

机器人决定（传感器和子程序）	一个机器人可以根据给定的条件，在指令的两种顺序之间进行“选择”。	把一个光传感器或者触感器联接到机器人大脑的正确端口。识别需要子程序的情景。写一个使用子程序的程序。
----------------	----------------------------------	---

每个活动都遵循一种相似的基本格式：（1）通过热身游戏用玩乐的方式介绍新的概念或者核心概念；（2）以搭建和/或编程任务强调这节课的核心概念；（3）编写在新环境下运用核心概念的小项目（以个人的或两人一组的形式）；（4）技术圈；（5）评估。

在完成六个触手可及方案课程的活动之后，到了让班上孩子创造一个最终作品的时候了。这是一个机会让学生重温先前学到的概念和技能，并把它们应用于自己选择的作品中。制作这些作品所需时间的长度随着课堂和教师的目标，以及课程要求而有所变化。最终的作品将放在一个开放的教室里，与更多社区人士分享。

儿童的触手可及方案的最终作品的实例包括机器人城市、动物会移动的公园、恐龙公园、马戏团、以及有着根据不同的传感器作出反应的机器人花的花园（Bers, 2008a）。这些作品都使用了廉价的回收材料。例如，在波士顿一个幼儿园的一个班的幼儿，在去过一座老城郊游之后，搭建了一个机器人自由之路，他们使用了硬纸盒来重现老城的历史性建筑物，并在盒子里装入了光传感器和马达，使建筑物充满了生机（Bers, 2008a）。

### 触手可及机器人技术概述

触手可及机器人技术方案开始探索的研究问题之一，就是“一个发展适宜性的机器人工具箱应该包含哪些元素，可以让幼儿获得一种成功的学习经验？”触手可及机器人技术方案是建立在近十年有关幼儿和机器人技术的研究成果的基础上（Bers, 2000; Cejka et al., 2006; Bers, 2008a）。在那段时间里，技术人员设计、实施和评估了不同的界面和技术（Martin et al., 2000），但并没有努力开发适合幼儿发展的界面。触手可及机器人技术方案利用了人机互动（HCT）领域的可触及的界面，这个界面通过以键盘或鼠标的方式，提供了可触及的系统，以克服写计算机程序的固有的缺陷（Blikstein, Buechley, Horn, & Raffle, 2010）。

一种可触及的程序语言，像以文本为基础的和以图形为基础的语言一样，是

一种用于向计算机发出指令的工具。双重形式的程序系统也是存在的。当使用以文本为基础的语言时，一个编程人员会使用像“开始”、“如果”和“重复”这样的词来写计算机指令。这样的编码必须根据严格的、经常令人沮丧的、句法的规则来书写。乐高就是针对儿童设计的，以文本为基础的编程语言的实例（Papert, 1980）。图像或视像语言用图代替了文本；程序通过鼠标对计算机屏幕上的图标进行排列并把它们连接起来而得以表现。Robolab（Rogers & Portsmore, 2004）和 SCRATCH（Resnick et al., 2009）就是为儿童设计的图形化的语言。这种语言仍然遵循句法的规则，但是可以通过一系列的图像将句法传输到电脑中。

通过可触及的语言，而不是依赖电脑屏幕上的图像和文本，用户通过摆弄和连接实物来建构计算机程序。可触及的语言利用了物体的物理特征（大小、形状、材质等。）来表现并执行句法。可触及的语言这一想法最初始于 20 世纪 70 年代中期（Perlman, 1976），并在近二十年后（Suzuki & Kato, 1995）得到了复兴。从那以来，世界各地的实验室开发出好几种可触及的语言（e.g. McNerney, 2004; Wyeth, 2008; Smith, 2007; Horn & Jacob, 2007）。

塔夫茨大学的研究人员已经采取了双重程序的方法，允许儿童在以屏幕为基础的和可触及的编程语言之间来回切换（Bers & Horn; 2010）。这个系统，叫做 CHERP（Creative Hybrid Environment for Robotic Programming 创造性的针对机器人编程的双重程序环境），允许儿童使用互锁的木质积木（图 3）进行编程。儿童通过使用积木的图形来代表机器人要执行的动作，可以在积木和屏幕程序之间来回切换。这种双重程序方法允许儿童用多种表征方法进行作业（Horn et al., in press）。



图 3. 塔夫茨大学开发的可触及的和屏幕上的 CHERP 语言元素

CHERP 使用一系列的图像—运行的技术以将实物程序传输到数字指令中去。语言中的每块积木都印刻着一块称为高级码号的环状标记 (Horn, Bers, & Jacob, 2009)。这些码号允许每条程序语句的位置、方向、大小、形状和类型都能很快地由数字图像来决定。一个标准的与台式计算机或笔记本电脑连接的摄像头把程序拍摄下来。一个编译程序把拍的照片转成数字编码, 这些数字编码是花了几秒中从机器人身上卸载下来的 (参见图 4)。

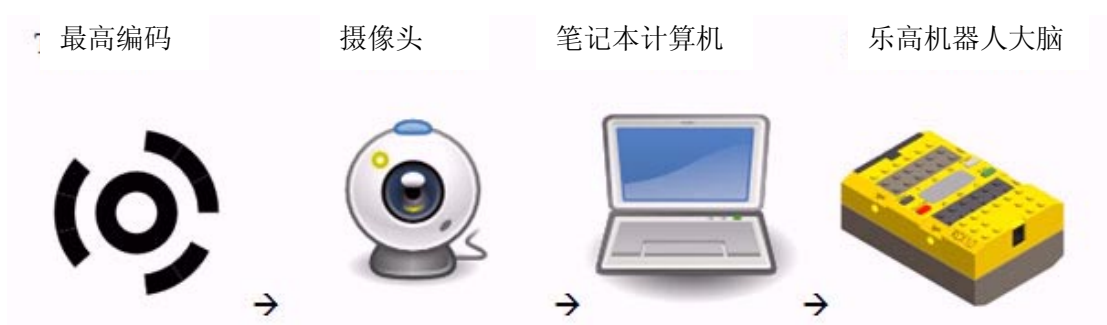


图 4. CHERP 下载过程

触手可及方案将 CHERP 与不同的机器人工具箱结合起来使用。儿童使用可触及的积木创造的程序, 并可以把它下载到乐高机器人的模板中。这个积木是一个嵌入式微型计算机或者“机器人脑”, 它置放在乐高头脑风暴工具箱中。塔夫茨大学的研究人员正在探索其他机器人工具箱的使用, 这些工具箱可能不包括乐高积木, 从而提供更高廉价的选择。这样的技术选择不在本文讨论的范围内。

### 与课程领域的联系

触手可及机器人方案的设计明显希望强调早期教育中的科学、技术、工程学和数学 (STEM) 中“缺失的中间两个字母”——T (技术) 和 E (工程学)。然而, 为了在课堂中顺利实施方案, 不管是从主题还是就学科概念和技能的角度, 必须综合并推进其他课程的内容 (Bers et al., 2002)。表 3 表明了触手可及方案的核心概念与其它学科领域知识之间可能的联系。



**表 3**  
触手可及方案的核心概念与跨学科领域之间的联系

核心概念	描述	学科联系
排序/流程控制	指令的顺序可以在一个程序中描述,并能通过机器人的行为按顺序表现出来。	<ul style="list-style-type: none"> <li>• 文学: 故事讲述、“如何做”大号图书</li> <li>• 符号系统</li> <li>• 设计可控的实验</li> <li>• 对几何学的基础探索</li> <li>• 因果联系</li> </ul>
循环	修正指令的顺序以使指令重复出现或者以受控的方式发生。	<ul style="list-style-type: none"> <li>• 进一步的几何学探索</li> <li>• 系统时间</li> <li>• 循环</li> <li>• 用符号传达信息</li> </ul>
参数	有些指令可以随着新的信息合理化。	<ul style="list-style-type: none"> <li>• 进一步的几何学探索</li> <li>• 数感</li> <li>• 计时和控制</li> </ul>
子程序	程序中有些指令提出问题,并根据答案让机器人做一件事或另一件事。	<ul style="list-style-type: none"> <li>• 因果联系</li> <li>• 逻辑</li> <li>• 进一步的科学探索</li> <li>• 执行功能技巧</li> <li>• 做出决定</li> </ul>
传感器	感知环境中的变化的装置,并将变化转成可以被观察	<ul style="list-style-type: none"> <li>• 自然界和人工制品</li> </ul>

	者、仪器或机器人识别的信号。	<ul style="list-style-type: none"> <li>生物学：动物和人的感知</li> </ul>
--	----------------	---

### 触手可及机器人方案与学习标准

触手可及机器人方案的课程所涉及的核心概念与国际技术教育协会（ITEA，2007）(<http://www.iteea.org/TAA/PDFs/xstnd.pdf>)和 2006 年马萨诸塞州的科技/工程学课程框架 <http://www.doe.mass.edu/frameworks/current.html>)提出的标准一致。这两组标准都强调儿童从学前班到二年级在自己已有知识的基础上学习工程学设计组件的重要性：

- 人们计划协助完成任务。（标准 2E； K-2）
- 每个人都能设计问题解决的方法。（标准 8A； K-2）
- 设计是一个创造性过程（以制成有用的产品和系统）。（标准 8B； K-2）
- 工程学设计过程包括发现问题、寻找主意、提出解决方法并与其他人进行分享。（标准 9A； K-2）
- 提出问题，进行观察，以帮助一个人发现事物是如何运作的。（标准 10A； K-2）

表 4 表明了概念和技巧之间的联系，根据 ITEA 标准和马萨诸塞州技术和工程学框架的要求，在课堂中应该强调这种联系。

表 4 触手可及方案的概念和技巧与 ITEA 标准和马萨诸塞州技术和工程学指南之间的联系		
核心概念	ITEA 标准	马萨诸塞州技术和工程学指南
搭建	<ul style="list-style-type: none"> <li>● 用废旧材料制作东西。（标准 2D； K-2）</li> <li>● 材料有许多不同的属性。（标准 2J； Gr 3-5）</li> <li>● 有的材料是可以回收再利用的。（标准 5A； K-2）</li> </ul>	<ul style="list-style-type: none"> <li>● 不管是人工还是天然的材料，自身的特征决定了它们的用途。（Gr PreK-2； 中心概念 1）</li> <li>● 发现并描述工具和材料的安全和正确的用法。（TE Gr PreK-2； Std 1.1）</li> <li>● 适宜的材料、工具和机器有助于我们解决问题和发明创造（和搭建）。（TE Gr 3-5）</li> </ul>

	<ul style="list-style-type: none"> <li>• 用设计过程搭建或建构一个物体。(K-2)</li> </ul>	(6-8); 中心概念 1)
机器人技术	<ul style="list-style-type: none"> <li>• 使用设计过程搭建或建构物体。(Std 11B; K-2)</li> <li>• 发现事物如何运作。(Std 12A; K-2)</li> <li>• 系统包含了能够协作共同完成一个目标的部分。(Std 2B; K-2)</li> <li>• 工具和机器运用能量进行运作。(Std 16D; Gr 3-5)</li> </ul>	<ul style="list-style-type: none"> <li>• 在教师的指导下,使用适宜的技术工具(……)来对问题进行界定和提出假设。(TL Gr 3-5; Std 3.6)</li> </ul>
编程	<ul style="list-style-type: none"> <li>• 识别并使用日常标记符号。(Std 12C; K-2)</li> <li>• 人们在用科技沟通时使用符号。(Std 17C; K-2)</li> <li>• 使用与其他学科相同的理念和技能研究科技。(Std 3A; K-2)</li> </ul>	<ul style="list-style-type: none"> <li>• 发现并解释符号和图形(……)是如何用来传达信息的。(TE Gr 6-8; Std 3.4)</li> </ul>
传感器	<ul style="list-style-type: none"> <li>• 自然界和人工制品是不同的。(Std 1A; K-2)</li> </ul>	<ul style="list-style-type: none"> <li>• 识别并描述自然界材料和人工材料的特点。(K-2)</li> <li>• 人类和动物使用身体的部位作为工具。(PreK-2)</li> </ul>

## 评估

早先讨论的 PTD 框架提供了设计教育方案和评估儿童的学习发展的准则。内容的创意和创造性(最开始的两个 C)的评估涉及到能力(或者理解的水平)和对该领域的信心。以下内容用来评估内容的创意和创造性:

- 学生的档案袋——由学生的设计日志、学生的编程样品(程序编码)和

学生的机器人技术作品组成。复杂的程度随着时间的推移发生变化，并对复杂性进行评估。

- 录像日志——在方案实施过程中至少进行 3 次录像（例如，方案开始时、中期和结束时），录像的内容是儿童展现他们一起努力所做的工作，并讲解他们的活动。
- 句法、语义和系统理解水平目录——为教师或研究者设定的一组问题，以在学期结束时评估每个儿童的理解（对句法的、语义的、或者系统的、以及在水平间的转换）水平（分为 1 至 5 级），用来评估课程中所有的学习目标，以及不同复杂水平下的每一个任务。下面的实例表明了第 2 部分的评估目录。在这一评估中，儿童给他们的机器人进行编码以随摇摇乐而舞，并引入“指令起作用的顺序”这一核心概念，这也是计算机思维中的一个关键概念。

第一部分：在列出的表述中，圈出与每个儿童相应的成就水平。如果儿童在活动中没有表现出所列的表述，就圈 NA。

5	4	3	2	1	0
在没有帮助的情况下完成任务。	在最少帮助下完成任务。	在定期帮助下完成任务。	在大量帮助下完成任务。	在一步一步指导下完成任务。	没有完成任务。
A.根据活动目标有目的地进行活动。					5 4 3 2 1 0 NA
B.根据自己预设的目标有目的地进行活动。如果是这种情况，目标是什么？					5 4 3 2 1 0 NA
如果根据自己预设的目标有目的地进行活动，他/她自己能够描述目标。					5 4 3 2 1 0 NA
把他/她的理念转换成机器人编码并使之行动。					5 4 3 2 1 0 NA
用句法正确的次序来组织积木或图形，写出一个活动功能正常的程序。					5 4 3 2 1 0 NA

A.通过阅读程序或者观察机器人执行程序，发现程序中不对的行为或者次序。	5 4 3 2 1 0 NA
B.对问题进行假设。	5 4 3 2 1 0 NA
C. 尝试解决问题。	5 4 3 2 1 0 NA
根据它们相应的行为选择正确的程序指令。	5 4 3 2 1 0 NA
机器人在受到控制和执行程序时，保持其核心部分的完整性。	5 4 3 2 1 0 NA
理解用 TUI 和 GUI 制成的程序必须通过电脑转化和传输给机器人。	5 4 3 2 1 0 NA

第二部分：向每个儿童提出问题，“当你制作一个程序时，你所放置的积木的不同顺序有不同的意义吗？”，从而为学生的日志和教师在学生制作程序和交流成果时的观察和谈话提供补充信息。依据以下标准标出儿童对如何给机器人编程的理解水平。

句法的	理解单个指令的功能，但不能理解如何选择并把指令组合起来以制作一个能够完成特定目标的功能性程序。
语义的	为程序选择适宜的指令并将其按正确的顺序放好。理解以特定的方式把部分合在一起会达成一个总体的成果。也许不能够制作一个完全满足特定目标的程序，或者当目标达到时并没有意识到。
系统	理解每个元素的功能，以及理解放置元素的顺序会导致某种特定的总体的结果。能够针对程序以正确的顺序给予正确的指令以达到特定的目标。

图 5.一个句法、语义和系统的目录

大多数儿童可能并不完全适合这里列出的类型，但可能处在向这些类型过渡的水平。目前的数据分析正被用于学习轨迹的建构。

儿童的合作和沟通是根据以下方式进行评估，通过分析儿童在一段时间的合作网络来评价他们的关怀和建立联系的水平，以及儿童在技术圈中的参与度(Lee & Bers, 2010)。

最后，社区共建和品行的选择是根据儿童在触手可及方案中总体上的参与程度和他对学习环境的贡献大小，特别是最后把作品放在开放室展览时来进行评估的。专家的徽章可看作儿童的个性品质的标志，并对随时间的变化进行了分析。

在触手可及方案中的评估与大多关注技术素养的方案不同，它不仅强调儿童技术经验中的认知维度，而且强调技术经验中的社会和道德维度，旨在帮助儿童以一种综合和全面发展的方式得到发展。

就研究而言，指导触手可及机器人技术方案的评估目标是双重的。第一个目标是采用与之前描述的积极技术发展（PTD）框架的组成部分相一致的方法，提供一个以事实为基础的系统的儿童学习和使用课程中所体现的核心概念的解释。第二个目标是建立这一学科领域中潜在的学习轨迹，该学习轨迹考虑了发展的进程和认识论的基础，以及建立逐步增加难度的特殊的活动或任务，这种难度与机器人技术核心概念和儿童的理解水平相适应（Clements & Sarama, 2004; Clements & Sarama, 2009）。对这一方案研究信息的介绍不属本文的探讨范围。与触手可及方案相关的进一步研究将会关注新的课程模块，实施更为廉价的使用日常材料的机器人技术系统，并为这一领域的学习轨迹建构一个有坚实基础的理论模型。

## 结论

这是一个对早期教育提出种种挑战的时代。联邦文件中提出的学业要求可能与日益受到关注的、对儿童发展需求的尊重相违背。为了提高我们国家年轻一代的技术素养，我们需要在早年开始推行教育。然而，特别是在技术和工程学方面，对儿童能够学习的内容缺乏必要的重新审视。我们也没有足够的研究基础来对儿童在创新科技方面到底能够学什么以及他们如何学进行评估。触手可及机器人技术方案发展的研究提供了一个研究的基础，它与把计算机科学和工程学理念引入早期课堂息息相关。到目前为止，触手可及方案的经验表明了当有年龄适宜性的技术、课程和教学法，幼儿就能够以与发展适宜性实践相一致的方式积极地参与计算机编程和机器人活动。

## 致谢

我真诚地感谢塔夫茨大学发展技术研究小组的我的学生：Jordan Crouser、Rachael Fein、Louise Flannery 和 Elizabeth Niro，他们在不同方面对这一方案做

出了贡献。我也感谢我以前的学生和现在西北大学工作的同事 Mike Horn，因为他在塔夫茨大学攻读博士学位时就设计了 TERN 并帮助我开始这项研究，也感谢我的来自人机交互实验室的同事罗 Robert Jacob 和来自塔夫茨大学 CEEO 的 Chris Rogers。最后，感谢国家科学基金会对本研究的支持（国家科学基金会授权文件 0735657 和国家科学基金会终身奖 IIS0447166）。本文的任何观点、发现和结论或建议不代表国家科学基金会的观点。

## 参考文献

Bers, Marina Umaschi, & Urrea, Claudia. (2000). Technological prayers: Parents and children exploring robotics and values. In Allison Druin & James Hendler (Eds.), *Robots for kids: Exploring new technologies for learning* (pp. 194-217). San Francisco: Morgan Kaufmann.

Bers, Marina Umaschi. (2006). The role of new technologies to foster positive youth development. *Applied Developmental Science, 10*(4), 200-219.

Bers, Marina Umaschi. (2008a). *Blocks to robots: Learning with technology in the early childhood classroom*. New York: Teachers College Press.

Bers, Marina U. (2008b). Engineers and storytellers: Using robotic manipulatives to develop technological fluency in early childhood. In Olivia N. Saracho & Bernard Spodek (Eds.), *Contemporary perspectives on science and technology in early childhood education* (pp. 105-125). Charlotte, NC: Information Age.

Bers, Marina Umaschi. (2010). When robots tell a story about culture...and children tell a story about learning (pp. 227-247). In Nicola Yelland (Ed.), *Contemporary perspective on early childhood education*. Maidenhead, UK: Open University Press.

Bers, Marina Umaschi, & Horn, Michael S. (2010). Tangible programming in early childhood: Revisiting developmental assumptions through new technologies. In Irene R. Berson & Michael J. Berson (Eds.), *High-tech tots: Childhood in a digital world* (pp. 49-70). Greenwich, CT: Information Age.

Bers, Marina; Doyle-Lynch, Alicia; & Chau, Clement. (in press). Positive technological development: The multifaceted nature of youth technology use towards improving self and society. In Cynthia Carter Ching & Brian Foley (Eds.), *Technology, learning, and identity: Research on the development and exploration of selves in a digital world*. Cambridge: Cambridge University Press.

Bers, Marina U.; Ponte, Iris; Juelich, Katherine; Viera, Alison; & Schenker, Jonathan. (2002). Teachers as designers: Integrating robotics into early childhood education. *Information Technology in Childhood Education Annual*, 123-145.

Blikstein, Paulo; Buechley, Leah; Horn, Michael; & Raffle, Hayes. (2010). A new age in tangible computational interfaces for learning. In K. Gomez, L. Lyons, & J. Radinsky (Eds.), *Proceedings of the 9th International Conference of the Learning Sciences* (Vol. 2, pp. 130-132). Chicago: International Society of the Learning Sciences.

Brosterman, Norman. (1997). *Inventing kindergarten*. New York: H.N. Abrams.

Cejka, Erin; Rogers, Chris; & Portsmore, Meredith. (2006). Kindergarten robotics: Using robotics to motivate math, science, and engineering literacy in elementary school. *International Journal of Engineering Education*, 22(4), 711-722.

Clements, Douglas H. (1999). The future of educational computing research: The case of computer programming. *Information Technology in Childhood Education Annual*, 147-179.

Clements, Douglas H., & Sarama, Julie (Eds.). (2004). Hypothetical learning trajectories. *Mathematical Thinking and Learning* [Special issue], 6(2).

Clements, Douglas H., & Sarama, Julie. (2009). *Learning and teaching early math: The learning trajectories approach*. New York: Routledge.

Craig, John J. (2005). *Introduction to robotics*. Upper Saddle River, NJ: Pearson Prentice Hall.

Damon, William. (2004). What is positive youth development? *Annals of the American Academy of Political and Social Science*, 591(1), 13-24.

Horn, Michael S.; Bers, Marina U.; & Jacob, Robert J. K. (2009, April). *Tangible programming in education: A research approach*. Paper presented at CHI '09. Boston, MA.

Horn, Michael S., & Jacob, Robert J. K. (2007). Designing tangible programming languages for classroom use. In Brygg Ullmer, Albrecht Schmidt, Eva Hornecker, Caroline Hummels, Robert J. K. Jacob, & Elise van den Hoven, *TEI '07: Proceedings of the First International Conference on Tangible and Embedded Interaction* (pp. 159-162). New York: Association for Computing Machinery.



Horn, Michael S.; Crouser, R. Jordan; & Bers, Marina U. (in press). Tangible interaction and learning: The case for a hybrid approach. *Personal and Ubiquitous Computing* [Special issue on tangibles and children].

International Technology Education Association (ITEA). (2007). *Standards for technological literacy: Content for the study of technology* (3rd ed.). Reston, VA: Author.

Kafai, Yasmin B., & Resnick, Michael. (1996). *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mahwah, NJ: Erlbaum.

Kantor, Rebecca; Elgas, Peggy M.; & Fernie, David E. (1989). First the look and then the sound: Creating conversations at circle time. *Early Childhood Research Quarterly*, 4(4), 433-448.

Kay, Alan C. (2003). Interview with Alan Kay. *Computers in Entertainment*, 1(1), 8.

Larson, Reed W. (2000). Toward a psychology of positive youth development. *American Psychologist*, 55(1), 170-183.

Lee, K. T., Bers, Marina. (2010). *Collaboration by design: Using robotics to foster social interaction in kindergarten*. Manuscript submitted for publication.

Lerner, Richard M.; Almerigi, Jason B.; Theokas, Christina; & Lerner, Jacqueline V. (2005). Positive youth development: A view of the issues. *Journal of Early Adolescence*, 25(1), 10-16.

Martin, Fred; Mikhak, Bakhtiar; Resnick, Mitchel; Silverman, Brian; & Berg, Robbie. (2000). To mindstorms and beyond: Evolution of a construction kit for magical machines. In Allison Druin & James Hendler (Eds.), *Robots for kids: Exploring new technologies for learning* (pp. 9-33). San Francisco: Morgan Kaufmann.

Massachusetts Department of Education. (2006, October). *Massachusetts science and technology/engineering curriculum framework*. Retrieved October 7, 2010, from <http://www.doe.mass.edu/frameworks/current.html>

McNerney, Timothy S. (2004). From turtles to tangible programming bricks: Explorations in physical language design. *Personal and Ubiquitous Computing*, 8(5), 326-337.

Papert, Seymour. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Papert, Seymour. (1993). *The children's machine: Rethinking school in the age of the computer*. New York: Basic Books.

Papert, Seymour. (2000). What's the big idea: Towards a pedagogy of idea power. *IBM Systems Journal*, 39(3-4), 720-729.

Pearson, Greg, & Young, Thomas A. (2002). *Technically speaking: Why all Americans need to know more about technology*. Washington DC: National Academy Press.

Perlman, Radia. (1976). Using computer technology to provide a creative learning environment for preschool children. *AI Memo 360: Logo Memo No. 24*. Cambridge, MA: MIT Artificial Intelligence Laboratory.

*Random House Webster's Unabridged Dictionary*. (2006). New York: Random House.

Repenning, Alexander; Webb, David; & Ioannidou, Andri. (2010). Scalable game design and the development of a checklist for getting computational thinking into public schools. *SIGCSE 2010: Proceedings of the 41st ACM technical symposium on computer science education*. New York: Association for Computing Machinery.

Resnick, Mitchel. (1996). New paradigms for computing, new paradigms for thinking. In Yasmin B. Kafai & Mitchel Resnick (Eds.), *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mahwah, NJ: Erlbaum.

Resnick, Mitchel. (2003). Playful learning and creative societies. *Education Update*, 8(6), Retrieved May 1, 2009, from <http://web.media.mit.edu/~mres/papers/education-update.pdf>

Resnick, Mitchel; Berg, Robbie; & Eisenberg, Michael. (2000). Beyond black boxes: Bringing transparency and aesthetics back to scientific investigation. *Journal of the Learning Sciences*, 9(1), 7-30.

Resnick, Mitchel; Maloney, John; Monroy-Hernández, Andrés; Rusk, Natalie; Eastmond, Evelyn; Brennan, Karen; Millner, Amon; Rosenbaum, Eric; Silver, Jay; Silverman, Brian; & Kafai, Yasmin. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11), 60-67.

Rinaldi, Carlina. (1998). Projected curriculum constructed through documentation—Progettazione: An interview with Lella Gandini. In Carolyn Edwards, Lella Gandini, & George Forman (Eds.), *The hundred languages of children: The Reggio Emilia approach—Advanced reflections* (2nd ed., pp. 113-126). Greenwich, CT: Ablex.

Rogers, Chris, & Portsmore, Merredith. (2004). Bringing engineering to elementary school. *Journal of STEM Education*, 5(3-4), 17-28.

Rogers, Chris B.; Wendell, Kristen; & Foster, Jacob. (2010). The academic bookshelf: A review of the NAE Report, "Engineering in K-12 education." *Journal of Engineering Education*, 99(2), 179-181. Retrieved October 6, 2010, from [http://findarticles.com/p/articles/mi\\_qa3886/is\\_201004/ai\\_n53931016/](http://findarticles.com/p/articles/mi_qa3886/is_201004/ai_n53931016/)

Rogoff, Barbara. (2003). *The cultural nature of human development*. New York: Oxford University Press.

Rogoff, Barbara; Turkanis, Carolyn Goodman; & Bartlett, Leslee. (2001). *Learning together: Children and adults in a school community*. New York: Oxford University Press.

Rusk, Natalie; Resnick, Mitchel; Berg, Robbie; & Pezalla-Granlund, Margaret. (2008). New pathways into robotics: Strategies for broadening participation. *Journal of Science Education and Technology*, 17(1), 59-69.

Scales, Peter C.; Benson, Peter L.; Leffert, Nancy; & Blyth, Dale A. (2000). Contribution of developmental assets to the prediction of thriving among adolescents. *Applied Developmental Science*, 4(1), 27-46.

Smith, A. C. (2007). Using magnets in physical blocks that behave as programming objects. In Brygg Ullmer, Albrecht Schmidt, Eva Hornecker, Caroline Hummels, Robert J. K. Jacob, & Elise van den Hoven, *TEI'07: Proceedings of the First International Conference on Tangible and Embedded Interaction* (pp. 147-150). New York: Association for Computing Machinery.

Suzuki, Hideyuki, & Kato, Hiroshi. (1995). Interaction-level support for collaborative learning: Algoblock—an open programming language. In John L. Schnase & Edward L. Cunnius (Eds.), *Proceedings of CSCL '95: The First International Conference on Computer Support for Collaborative Learning* (pp. 349-355). Mahwah, NJ: Erlbaum.

Theokas, Christina, & Lerner, Richard M. (2006). Observed ecological assets in families, schools, neighborhoods: Conceptualization, measurement, and relations with positive and negative developmental outcomes. *Applied Development Science*, 10(2), 61-74.

Turbak, Franklyn, & Berg, Robbie. (2002). Robotic design studio: Exploring the big ideas of engineering in a liberal arts environment. *Journal of Science Education and Technology*, 11(3), 237-253.

Turkle, Sherry, & Papert, Seymour. (1992). Epistemological pluralism and the reevaluation of the concrete. *Journal of Mathematical Behavior*, 11(1), 3-33.

Wang, X. Christine, & Ching, Cynthia Carter. (2003). Social construction of computer experience in a first-grade classroom: Social processes and mediating artifacts. *Early Education and Development*, 14(3), 335-361.

Wing, Jeannette M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. Retrieved June 10, 2010, from <http://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing06.pdf>

Wyeth, Peta. (2008). How young children learn to program with sensor, action, and logic blocks. *International Journal of the Learning Sciences*, 17(4), 517-550.

## 作者信息

Marina Umaschi Bers 博士是塔夫茨大学 Eliot-Pearson 儿童发展和计算机科学系的副教授。她的研究涉及提高积极儿童发展的创新学习技术的设计和研发。在塔夫茨大学，Bers 教授主持了跨学科发展技术研究小组。Bers 教授获得了 2005 年的美国青年科学家最高荣誉（(PECASE)，是美国政府给在事业早期的杰出研究者的最高荣誉。她也获得了一个五年的国家科学基金会（NSF）年轻研究者事业奖，以及颁给在人文研究以及学习技术领域的早期就做出突出贡献的美国教育研究协会（AERA）简·霍金斯奖。在过去 14 年中，Bers 教授已经孕育并设计了不同的技术工具，包括从机器人技术到虚拟世界的工具。她在校外教育、博物馆和医院中进行了研究，也在美国国内外进行了研究。她的书《从积木到机器人：早期课堂中的技术学习》由教师学院出版社于 2008 年出版。她的即将出版的书《新技术的积极使用：针对幼儿学习和发展的数字经验的设计》将于 2012 年在牛津大学出版社出版。Bers 博士来自阿根廷，本科毕业于布宜诺斯艾利斯大学的社会传媒专业。在 1994 年，她来到美国，在这儿她获得了波士顿大学的教育媒体和技术学的硕士学位和麻省理工学院新闻实验室的科学硕士和博士学位，并与 Seymour 共事。对于更多的 Bers 的信息，请参见 <http://www.tufts.edu/~mbers01/>。

Marina U. Bers, Ph.D.  
Associate Professor  
DevTech Research Group  
Eliot Pearson Department of Child Development  
Computer Science Department

Tufts University  
105 College Ave  
Medford, MA 02155  
Telephone: 617-627-4490  
Fax: 617-627-3503  
Email: [Marina.Bers@tufts.edu](mailto:Marina.Bers@tufts.edu)

译者：中国华东师范大学学前教育系 程阳春

审校：中国华东师范大学学前教育系 周欣