

基于隐喻的机器人编程方法：促进幼儿的计算思维和积极学习行为

A metaphor-based robot programming approach to facilitating young children's computational thinking and positive learning behaviors

董倩 1, 陈禹辰 2, 贾若文 1, 张新立 1*

1 温州大学教育学院

2 悉尼大学教育学院

zhxl@wzu.edu.cn

【摘要】 在人工智能时代，培养幼儿的计算思维（CT）尤为重要。可编程机器人为幼儿提供了学习编程和培养计算思维的绝佳机会。然而，幼儿在学习抽象的计算思维概念时可能存在困难。因此，本研究提出了一种基于隐喻的机器人编程（MRP）方法，并探讨了其对幼儿 CT 和行为模式的影响。本实验共招募了 118 名 5-6 岁的儿童，实验组采用基于隐喻的机器人编程（MRP）方法，对照组采用传统的机器人编程（CRP）方法。结果显示，采用 MRP 方法可以促进学生的计算思维、学习成绩以及更积极的学习行为。因此，采用基于隐喻的机器人编程（MRP）方法有助于幼儿编程教学和 CT 发展。

【关键词】 教学/学习策略；改进课堂教学；交互式学习环境；隐喻；计算思维

Abstract: In the era of artificial intelligence, cultivating computational thinking (CT) in young children is particularly important. Programmable robots provide an excellent opportunity for young children to learn programming and develop computational thinking. However, young children may encounter difficulties when learning abstract CT concepts. Therefore, this study proposes a metaphor-based robotics programming (MRP) approach and explores its impact on young children's CT and behavioral patterns. A total of 118 children aged 5-6 were recruited for this experiment; the experimental group adopted the MRP method, while the control group used traditional robotics programming (CRP). The results show that the MRP method can promote students' computational thinking, academic performance, and more positive learning behaviors. Thus, using a metaphor-based robotics programming (MRP) approach can contribute to young children's programming education and CT development.

Keywords: Teaching/Learning Strategies, Improving Classroom Instruction, Interactive Learning Environments, Metaphor, Computational Thinking

1. 提出问题

在人工智能时代，计算思维（Computational Thinking, CT）作为一项数字能力受到了极大关注（Bers, 2019）。计算思维是一种思维能力，它利用计算机科学的基本概念来解决技术问题和现实生活中的问题（Wing, 2006）。最近的研究表明，早期接触 CT 可以预测 STEM 学科的学习、执行功能、21 世纪能力、甚至是未来的职业道路（Sun et al., 2022; Bati, 2021）。幼儿期是儿童成长和学习的阶段（Piaget, 1953），因此培养幼儿的 CT 至关重要。

编程被认为是促进学习者 CT 的最主要、最有效的手段（Bers, 2018; Macrides et al., 2022）。可编程机器人作为一种无屏幕且适合儿童发展的手段，为低年级儿童提供了接触编程和 CT 的绝佳会（Critten et al., 2021）。然而，传统教学方法侧重于抽象概念介绍，未充分结合具体经验，导致幼儿难以掌握如算法、循环等抽象概念（Relkin et al., 2021）。

概念隐喻理论（CMT）指出，隐喻能让个体将相对抽象的经验领域与我们熟悉的更简单或更清晰的经验领域联系起来（Lakoff, 1993）。之前的研究已经验证了在语言、数学和生物等多学科教学中使用隐喻的有效性（Reinhardt, 2020; Yee, 2017），并在编程教学中显示潜力（Pérez-Marín et al., 2020）。

因此，本研究提出一种基于隐喻的机器人编程（MRP）方法来培养幼儿的 CT 能力。此外，本研究进一步探讨了 MRP 方法对幼儿行为模式的影响。我们假设，采用 MRP 方法的幼儿能更好地提高 CT 能力，并表现出更积极的学习行为。本研究的主要研究问题如下：

- (1) 与传统的机器人编程（CRP）方法相比，MRP 方法是否能更好地促进幼儿的 CT？
- (2) MRP 方法比 CRP 方法更能促进幼儿的积极学习行为吗？

2. 开发基于隐喻的机器人编程(MRP)方法

2.1. MRP 学习环境

本研究使用 Matatalab 编程套件作为学习环境(图 1)。Matatalab 是专为 4-9 岁儿童设计的无屏幕可编程机器人套件，包括一个可编程机器人(MatataBot)、指挥塔、控制板和编程块。Matatalab 编程套件可帮助儿童学习基本的编程和 CT 概念，探索机器人编程问题解决任务，并发展 CT(Fu et al., 2023; Yang, Ng, & Gao, 2022)。



图 1 Matatalab 编程工具包

2.2. 采用 MRP 方法的课程设置

机器人编程课程包括 10 个 40 分钟的活动，每周一个活动。孩子们被分成两人一组用一个 Matatalab 编程工具包。该课程名为“自然之旅”。该课程的主要学习目标是(1)让孩子们理解并掌握 CT 概念，包括硬件、软件、算法、模块化、调试等；(2)通过学习和练习，让孩子们掌握基本编程模块的符号表示法:前进、后退、左转、右转、数字、循环等；(3)让孩子们将不同的编程模块结合起来，运用基本的 CT 概念解决问题；(4)让孩子们体验机器人编程的乐趣。

2.3. MRP 方法的教学过程

MRP 方法的教学过程主要包括六个阶段:(1)根据目标域设计源域的事件和活动；(2)介绍源域的事件和活动；(3)通过源域解释目标域；(4)介绍与目标域相关的编程模块；(5)发布与目标域相关的机器人编程任务；(6) 分享和反思(Bers, 2019)。

第一阶段是根据目标域设计源域的事件和活动。隐喻的基本要素是目标域、源域和映射，从源域到目标域的映射，就是把源域映射到目标域上，理解源域的结构来理解目标域 (MacCormac, 1990; Saban, 2006)。在这一阶段，教师确定机器人编程学习所涉及的抽象 CT 概念(目标域)，并根据目标 CT 概念的特点设计适当的、以学习者为中心的源域事件和活动。源域的设计原则是应与儿童的日常生活情境紧密联系，他们已经学习并清楚理解的现有知识。本研究组建了一个专家小组，包括两名教育技术研究人员和两名经验丰富的幼儿 STEM 教师，以讨论和决定源域的事件和辅助活动。表 1 显示了本研究设计的目标域和源域概览。

表 1 本研究设计的目标域和源域概况

目标域	源域
算法/序列:通过一系列连续的步骤逻辑性地解决问题。	洗衣服的步骤:通过“洗衣服的步骤”这个比喻来解释算法/序列的概念:一个男孩在踢完足球后发现衣服弄脏了，遇到一个大问题。因此，他想洗脏衣服。他会按步骤洗吗?他应该一步一步地洗:脱掉衣服，打开洗衣机门，等待一段时间，拿到干净的衣服……就像这个男孩按照特定的步骤洗衣服一样，算法/序列是按照逻辑顺序一步一步地解决特定问题。
表征:理解符号代表概念、指令等。	交通灯和路标的指令被用作比喻来教授表示概念。当驾驶汽车遇到道路上的交通灯/路标时，交通灯/路标会向我们发出信号或指示。每个信号或指示代表一个特定的命令。我们需要识别交通灯/路标的命令，如停止、前进、左转和右转，然后遵循这些命令。同样，当我们遇到程序中的指令时，我们需要识别特定的命令或动作，并遵循它们。
循环:控制结构的一个相关概念，指的是重复执行同一指令多次，直到达到某个条件，有时等同于重复。	选择旋转木马作为比喻来解释循环的概念:当我们去主题公园时，旋转木马会一次又一次地旋转;例如，它不会停止，直到达到 20 次的条件。同样，循环会一次又一次地重复相同的命令，直到满足条件。
调试:识别错误的部分并尝试解决和修复它。	修理坏汽车的比喻被用来说明调试的概念。当汽车不能正常工作，我们需要找到坏的部分，思考原因，并尝试修理，直到它恢复正常。就像这样，调试是尝试找到错误并对其进行改进的过程，直到正确为止。

第二阶段是介绍源域的事件和活动。在这一阶段，教师通过介绍源域的事件和活动让幼儿回忆和感知与日常生活情境或熟知的知识相关的概念。然后，教师引导幼儿提取源域的特征，为接下来学习抽象的 CT 概念打下坚实的基础。以循环概念为例。首先，教师在屏幕上展示了旋转木马的图片，并播放了一段视频引导学生了解旋转木马的工作原理和特点(图 2)，并进行引导互动。在与孩子们互动后，教师绘声绘色地说道：“当我们在主题公园乘坐旋转木马时，它会反复旋转，直到达到 10 次的条件才会停止。”



图 2 教师向学生介绍旋转木马的事件和活动

第三阶段是通过源域解释目标 CT 概念。在这一阶段，教师引导孩子们注意到源域与目标 CT 概念之间的相似性，从而帮助他们更好地理解 and 掌握抽象概念(Peelle, 1983)。以循环概念为例。首先，教师向全班介绍了循环概念：“我们刚刚学习了旋转木马的工作原理。与旋转木马的工作原理一样，循环也是一遍又一遍地重复相同的指令，直到满足某个条件为止。”随后，教师引导幼儿注意旋转木马(源域)和循环(目标域)之间的相似之处。例如，一名儿童回答道：“旋转木马重复旋转，而循环也是重复执行一个动作。”

第四阶段涉及引入与目标领域相关的符号表征。每个编程块代表一个特定的指令。在这一阶段，教师向全班介绍与目标 CT 概念相关的编程指令/块，并结合案例解释其功能和用途。以循环为例。首先，教师向全班展示了循环程序块：“在 Matatalab 编程工具包中有相应的循环编程块:循环开始和循环结束(图 3)。”然后，教师通过案例解释了它们的功能和用途：“我们学过循环就是重复执行相同的指令，直到满足条件为止，你能认出这段代码中重复的指令吗?它们重复了多少次(图 4)?”孩子回答这个问题后，教师向全班解读了它们的功能和用法。



图 3 教师向学生介绍循环编程块 图 4 教师通过案例介绍编程块

第五阶段是针对目标域发布机器人编程任务。在这一阶段，教师针对 CT 概念的目标域发布机器人编程任务，并向儿童明确任务的要求和目标。孩子与同伴运用目标 CT 概念合作设计方案、编写、执行并调试程序，最终完成编程任务。教师观察孩子们的表现，并在必要时提供帮助。再以循环概念为例：教师发布了相应的编程问题解决任务：“玛塔四周转一转(图 5)。我们刚刚学习了循环，并将利用循环来解决机器人编程任务。请帮助玛塔实现目标。”



图 5 教师给学生布置机器人编程任务

第六阶段是分享与反思。在这一阶段，教师组织孩子们分享自己解决问题的方案和机器人编程作品，随后给予评价反馈并对本节课进行总结。接下来，孩子们对自己的方案和机器人编程作品进一步完善。

3.实验设计

3.1. 研究对象

本研究采用了准实验设计。参与者的纳入标准是他们以前没有相关的培训经验。研究从华东地区一所公立幼儿园自愿招募了 118 名 5-6 岁的儿童(平均=5.71 岁)。随机分配 60 名儿

童(32名男孩和28名女孩)为实验组,58名儿童(29名男孩和29名女孩)为对照组。实验组采用MRP方法,对照组采用CRP方法。人口统计学变量(即年龄和性别)无明显差异, $P>.05$ 。

3.2. 实施保真度

在本研究中,我们努力确保高度的实施保真度:首先,两组均由同一位经验丰富、训练有素的教师授课。教师参加了三轮为期2小时的专业培训,以熟悉机器人编程工具、课程内容、隐喻教学过程等;其次,要求教师每周进行反思;最后,对两组的每节课进行录像,并由主要研究人员进行评估。

3.3. 测量工具

3.3.1. 计算思维测试

为了评估的CT能力,我们改编了用于不插电的CT测试的工具——TechCheck-K。它是由Rekin和Bers(2021年)根据六个维度,即硬件/软件、算法、表征、模块化、控制结构和调试,专门为幼儿开发的CT测试工具,其Cronbach's α 为0.78,达到了良好的内部信度。

3.3.2. 儿童学习行为编码方案

本研究根据代表性的文献(Sun et al., 2021),归纳出初步的儿童编程学习行为编码框架。接着对两组儿童的小规模样本进行了初步审查,检查是否捕捉到了所有行为特征。然后,专家小组就编码方案的适用性进行了讨论,并根据反馈意见对编码方案进行了完善。二位评分者之间的Kappa值为0.82,表明一致性可以接受。表2显示了儿童编程学习行为的编码方案。

表2 儿童编程学习行为的编码方案

代码	行为	说明
LT	听老师讲课	孩子们在课堂上听老师讲课。
IT	与教师互动	孩子们回答问题或向老师反馈。
DP	与同伴讨论	孩子们与伙伴讨论学习活动、编程和意见。
WP	编写程序	孩子们规划路线,摆放编程积木。
EP	执行程序	孩子们按下启动按钮,执行程序。
AH	寻求帮助	孩子们向老师或其他小组成员寻求帮助。
DE	调试	孩子们检查工具、编程块、步骤或路线,并尝试改进他们。
ET	扩展任务	孩子们在完成既定任务后,会自己尝试其他解决方案或者拓展任务。
SH	分享	孩子与教师或同伴分享他们的解决方案或任务。
IB	无关行为	孩子们聊天、玩耍或表现出其他不相关的行为。

3.4. 实验流程

在招募参与者后,儿童需要完成15分钟的CT前测。随后,在幼儿园科学教室开展了为期10周的机器人编程课程(每周40分钟)。实验组采用MRP方法,将隐喻方法融入机器人编程教学。对照组则采用CRP方法,侧重于传统的讲授的机器人编程教学。实验结束后,两组均立即进行了CT后测。

3.5. 数据分析

针对问题1,我们进行了单因素方差分析(ANCOVA),通过控制两组幼儿的CT前测得分,研究两组幼儿CT后测的差异。所有数据均使用SPSS26.0软件进行分析。

针对问题2,研究人员对幼儿的课堂表现进行录像,并特意选取了后期幼儿对所采用的教学方法已经比较熟悉的两节活动课对两组幼儿的学习行为进行编码。当儿童做出特定行为时,研究人员会将其记录下来。研究人员通过GSEQ软件对儿童的学习行为进行了滞后序列

分析(LSA)。Z 值大于 1.96 代表关系显著。然后，绘制学习行为转换图来比较两组的行为模式。

4.结果分析

4.1. 儿童计算思维分析

Shapiro-Wilk 检验的结果表明，本研究收集的所有数据都近似于正态分布($p>.05$)。

在控制 CT 前测得分的情况下，对儿童的后测得分进行了方差分析比较(表 3)。Levene 方差齐性检验的结果($F=1.39, p=0.242>0.05$)显示无显著性水平，说明满足了组间方差齐性假设。因此，可以进行方差分析。方差分析结果显示，在控制了儿童测试前的 CT 分数后，儿童测试后的 CT 分数存在显著差异($F=40.29, p<0.001, \eta^2=0.26$)。实验组儿童测试后 CT 分数的调整后均值和标准误差为 11.98 和 0.19，而对照组则分别为 10.33 和 0.18。研究发现，实验组的 CT 测试后得分高于对照组，这表明 MRP 组在 CT 方面的表现优于 CRP 组。

此外，还对 CT 的各个维度进行了方差分析。结果显示，两组学生在 CT 的四个维度上存在显著差异，即算法、序列($F=26.80, p<0.001, \eta^2=0.17$)、表征($F=5.90, p<0.05, \eta^2=0.04$)、控制结构($F=5.17, p<0.05, \eta^2=0.04$)和调试($F=20.28, p<0.001, \eta^2=0.11$)，这意味着 MRP 方法比 CRP 方法更有利于促进儿童学习算法/序列、表征、控制结构(包括循环)和调试。

表 3 实验组和对照组的测验后 CT 分数的方差分析结果

维度	组别	N	Mean	SD	Adjusted Mean	Std.error	F	η^2
整体	实验组	60	11.95	1.48	11.98	0.19	40.29***	0.26
	对照组	58	10.36	1.77	10.33	0.18		
硬件/软件	实验组	60	1.67	0.48	1.66	0.06	0.07	0.00
	对照组	58	1.64	0.48	1.64	0.06		
算法/排序	实验组	60	3.77	0.87	3.73	0.12	26.80***	0.17
	对照组	58	2.84	1.04	2.88	0.12		
表征	实验组	60	1.37	0.55	1.30	0.06	5.90*	0.04
	对照组	58	1.02	0.44	1.08	0.06		
模块化	实验组	60	1.52	0.50	1.54	0.06	0.56	0.00
	对照组	58	1.50	0.50	1.48	0.06		
控制结构	实验组	60	1.88	0.32	1.91	0.04	5.17*	0.04
	对照组	58	1.79	0.41	1.77	0.04		
调试	实验组	60	1.75	0.44	1.83	0.05	20.28***	0.11
	对照组	58	1.57	0.50	1.49	0.05		

4.2. 儿童学习行为模式分析

实验组共收集到 998 个事件代码，对照组收集到 968 个事件代码。在 MRP 实验组中，最常见的行为是 WP(即编写程序，15.6%)、DP(即与同伴讨论，15.2%)和 LT(即听老师讲课，14.0%)。对照组儿童的 IB(即无关行为，16.5%)、WP(即编写程序，12.4%)和 DP(即与同伴讨论，11.8%)表现较多。相比之下，对照组出现了更多的无关行为(如聊天分散注意力或玩耍)。

根据表 4 和表 5 所示的两组调整后残差表，MRP 实验组有 16 个显著序列，而对照组有 23 个显著行为序列。

两组重要行为序列的过渡矩阵图分别如图 6 和图 7 所示。黑线表示两组同时出现的重要序列，而红线表示只存在于实验组或对照组的重要序列。如图 6 所示，在教学过程中，实验组幼儿反复聆听教师的讲解并与教师互动(LT→IT→LT)，然后与同伴讨论(LT→DP)。在与同

伴讨论后，孩子们尝试编写程序，然后较少的孩子会寻求帮助(WP→AH)，而较多的孩子会执行程序(WP→EP)。当儿童完成执行程序后，一些儿童调试然后分享作品(EP→DE→SH)，而另一些儿童则直接分享作品(EP→SH)。实验组的孩子们反复分享和扩展学习任务(SH→ET→SH)。这表明，在实验组中使用隐喻学习编程有助于儿童掌握基本概念和技能，因为他们无需太多帮助就能完成编程作业，甚至能自发地扩展任务。

从图 7 中可以发现，对照组的儿童除了参与教学过程的儿童(LT→IT→LT)外，都会出现一些不相关的行为(IT→IB→LT)，这可能是因为有些儿童不能理解抽象的概念。在编写程序的环节中，大多数儿童喜欢与同伴反复讨论(WP→DP)，也有一些儿童会执行程序(WP→EP)或寻求帮助(WP→AH)。之后，他们往往会在执行程序(AH→EP→AH)或调试(AH→DE→AH)过程中反复请求他人帮助。这表明，对照组可能很难在没有他人帮助的情况下完成编程任务。此外，孩子们会在调试或扩展任务(DE→SH, ET→SH)后进行分享。此外，在扩展和分享任务后会做出一些不相关的行为(ET→IB, SH→IB)。

表 4 实验组调整后的残差值

Given	LT	IT	DP	WP	EP	AH	DE	ET	SH	IB
LT	-5.05	14.62*	4.93*	-1.91	-4.26	-2.20	-2.55	-3.38	-4.35	1.23
IT	17.85*	-3.00	-2.70	-3.52	-2.79	-1.44	-1.67	-2.21	-2.85	-0.43
DP	-3.77	-3.41	-3.95	15.07*	-3.18	-1.64	-1.90	-2.02	0.81	0.88
WP	-3.88	-3.52	1.41	-3.80	17.70*	4.80*	-1.96	-2.60	-3.35	-2.88
EP	-3.08	-2.79	1.66	-3.28	-2.60	2.09*	8.90*	-0.90	5.36*	-0.68
AH	-1.59	-1.44	-0.94	4.08*	-1.34	-0.69	7.22*	-1.06	-1.37	-1.18
DE	-1.80	-1.63	-1.88	-1.91	-1.52	1.96	-0.91	-0.27	8.99*	0.37
ET	-1.42	-1.28	-0.67	-1.51	-1.20	-0.62	-0.72	-0.95	9.13*	-1.05
SH	-2.28	-2.07	-2.39	-2.43	-1.93	-0.99	-1.15	12.67*	-1.97	5.85*
IB	5.28*	-2.45	1.25	-1.98	-2.28	-1.18	-1.37	5.31	-1.28	-2.00

* $Z > 1.96$

表 5 对照组调整后的残差值

Given	LT	IT	DP	WP	EP	AH	DE	ET	SH	IB
LT	0.00	11.58*	-0.89	2.33*	-4.09	-3.29	-3.45	-2.99	-2.64	0.02
IT	7.64*	0.00	-3.09	0.03	-2.85	-2.29	-2.40	-2.08	-1.84	3.60*
DP	-3.21	-3.09	0.00	5.16*	8.53*	-2.32	-0.40	-2.10	1.99*	-3.87
WP	-3.32	-3.20	7.96*	0.00	3.01*	2.74*	-2.51	-2.17	-1.92	-1.27
EP	-2.96	-2.85	-0.69	-2.98	0.00	7.45*	6.97*	-1.94	-1.03	0.18
AH	-2.24	-2.16	0.60	-2.25	3.34*	0.00	9.96*	-1.46	-1.29	-2.70
DE	-2.32	-2.24	-2.26	-2.34	-1.51	3.85*	0.00	7.52*	5.38*	-0.96
ET	-1.62	-1.56	-1.58	-1.63	-1.45	-1.17	-1.23	0.00	8.36*	3.76*
SH	-1.35	-1.30	-1.31	-1.36	-1.21	-0.97	-1.02	4.08*	0.00	4.43*
IB	6.16*	-0.83	-0.52	-0.35	-3.23	-2.60	-2.72	4.97*	-1.50	0.00

* $Z > 1.96$

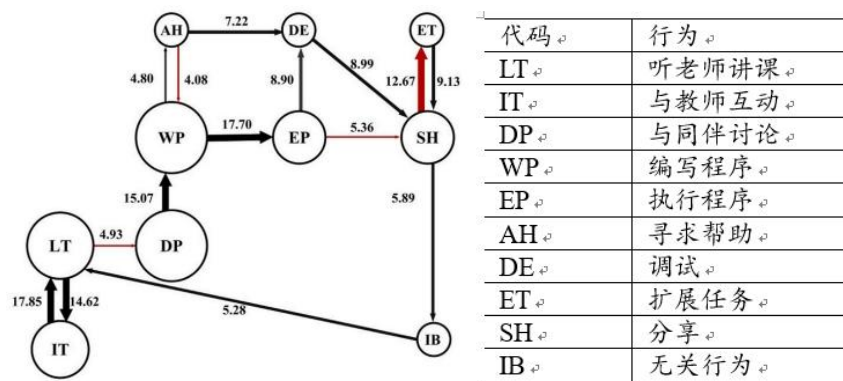


图 6 实验组重要行为序列的过渡图

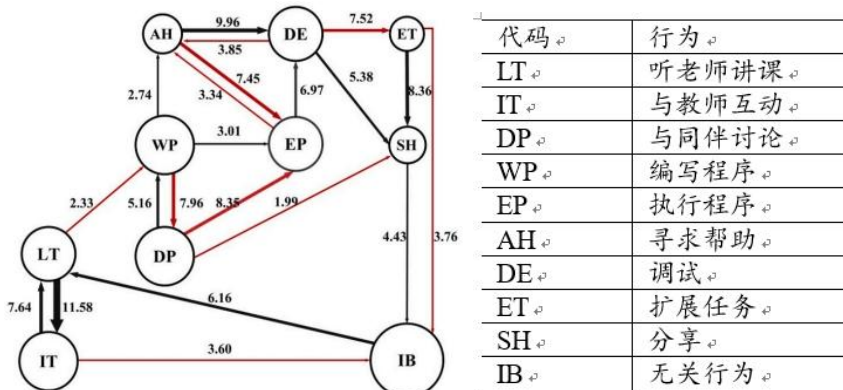


图 7 对照组重要行为序列的过渡图

此外，实验组有四种行为模式达到了显著水平。LT→DP，即孩子们在听课后愿意与同伴讨论(LT→DP)。AH→WP表示他们在寻求帮助后会继续编写程序。最理想的情况是EP→SH和SH→ET:孩子们会在执行程序后转而分享他们的作品，而不是寻求帮助或调试，或者他们会在分享后扩展他们的学习任务。对照组中有 10 对达到显著水平的序列。LT→WP代表孩子们先听课后编写程序。WP→DP、DP→EP和DP→SH表示儿童在编写程序后与同伴讨论，然后执行程序或进行分享。DE→ET表示儿童调试并扩展任务。不过，也有一些不足之处:AH→EP和EP→AH表示幼儿会反复寻求帮助并执行程序;IT→IB和ET→IB表示幼儿在与教师互动或扩展任务后会做出与任务无关的行为。

5.讨论和结论

研究采用准实验设计，探讨了所提出的“隐喻式机器人教学法”对 CT 和行为模式的影响。结果显示，参加 MRP 组的儿童在 CT 方面的表现优于 CRP 组。此外，行为分析表明，提出的 MRP 方法可促进儿童取得优异的学习成绩和更积极的行为，从而帮助他们实现学习目标。

针对研究问题一，实验组在整体 CT 方面的表现优于对照组，并且在算法、序列、表示法、控制结构和调试方面的表现优于对照组。这表明与 CRP 方法相比，MRP 方法能更好地提高幼儿的 CT 能力。这一发现与 Pe'rez:Marin 等人(2020)的研究结果一致，即在编程学习中提供隐喻可以增强学习者的 CT 能力。从理论上讲，隐喻是一种有益的教学工具，能够将抽象的概念与儿童熟悉的生活经验联系起来，帮助学习者从已知概念迁移到未知概念，更好地促进学习者的学习成绩和学习表现。在本研究中教师通过旋转木马等生活实例来解释循环概念，使孩子们更容易理解并应用到机器人编程任务中，从而发展他们的 CT。皮亚杰的认知

发展理论指出,前运算阶段儿童的概念知识建构依赖于更具体、更熟悉的经验来建构抽象概念(Piaget, 1953)。因此,如果能在机器人编程教学中提供隐喻,将能有效促进幼儿 CT 的发展。

针对研究问题二,滞后序列分析显示,MRP 组的儿童与 CRP 组表现出不同的行为模式。实验组儿童表现出以下特点:(1)更专注于教学过程;(2)更投入地完成编程任务;(3)更多儿童在完成规定任务后投入到拓展学习任务中。这可能是因为在教授儿童机器人编程时,提供隐喻可以帮助教师建立一个抽象的 CT 概念与生动具体的内容之间的联系,使得儿童可以很好地感知、理解和建构相应的 CT 概念,并将其迁移和应用于完成机器人编程任务(Pérez-Marín et al., 2018, 2020)。此外,从更多儿童在完成规定任务后表现出主动扩展学习任务的积极行为来看,MRP 方法可以丰富儿童的编程学习体验,激发儿童的技能迁移能力和发散思维。至于 CRP 对照组,也可以得出三个主要结论:(1)对解决问题的任务更加茫然;(2)在调试程序时更多地寻求帮助;(3)课堂上表现出更多的无关行为,这与詹晓明等人(2022)的发现相似。这反映了传统机器人编程教育中存在的问题,即如果没有提供适当的教学方法,儿童可能难以理解和掌握基本概念,并试图寻求帮助或在学习任务中表现不佳。因此,所建议的基于隐喻的机器人编程方法可促进儿童取得优异的学习成绩和更积极的行为,从而帮助他们实现学习目标。

6. 结语

本研究提出基于隐喻的机器人编程方法,并通过实验研究证实了其对 CT 和行为模式的积极影响,在编程和 CT 教育的教学方法方面做出了贡献。此外,本研究仍存在一些局限性。首先,本研究的参与者来自华东地区的一所幼儿园,这可能不具有普遍适用性。其次,本研究只实施了有限的干预,没有评估干预的长期效果。基于此,我们对后续研究提出了几点建议:扩大样本多样性和年龄范围;考虑个体差异(如性别、编程经验等);如何提升自我调节和参与度,减少无关行为;继续开发有效教学法。

参考文献

- Bati, K. (2022). A systematic literature review regarding computational thinking and programming in early childhood education. *Education and Information Technologies*, 27(2), 2059-2082.
- Beilin, H. (1971). Developmental stages and developmental processes. *Measurement and piaget*, 172-188.
- Bers, M. U., González-González, C., & Armas-Torres, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education*, 138, 130-145.
- Lakoff, G. (1993). The contemporary theory of metaphor.
- Milner, W. W. (2010). A broken metaphor in Java. *ACM SIGCSE Bulletin*, 41(4), 76-77.
- Peelle, H. A. (1983). Computer metaphors: Approaches to computer literacy for educators. *Computers & Education*, 7(2), 91-99.
- Pérez-Marín, D., Hijón-Neira, R., Bacelo, A., & Pizarro, C. (2020). Can computational thinking be improved by using a methodology based on metaphors and scratch to teach computer programming to children?. *Computers in Human Behavior*, 105, 105849.

- Reinhardt, J. (2020). Metaphors for social media-enhanced foreign language teaching and learning. *Foreign Language Annals*, 53(2), 234-242.
- Relkin, E., de Ruiter, L. E., & Bers, M. U. (2021). Learning to code and the acquisition of computational thinking by young children. *Computers & education*, 169, 104222.
- Saban, A. (2006). Functions of metaphor in teaching and teacher education: A review essay. *Teaching education*, 17(4), 299-315.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Yang, W., Ng, D. T. K., & Gao, H. (2022). Robot programming versus block play in early childhood education: Effects on computational thinking, sequencing ability, and self-regulation. *British Journal of Educational Technology*, 53(6), 1817-1841.