The Design Space of Tangible Interfaces for Computational Tinkerability

Krithik Ranjan[1], Anika Mahajan[1], Rishi Vanukuru[1], and Ellen Yi-Luen Do[1]

[1] ATLAS Institute, University of Colorado Boulder, USA

Abstract

Tangible interfaces offer a powerful approach for engaging learners in computational experiences, fostering intuitive, collaborative, and constructionist-driven learning. This work explores the landscape of tangible interfaces for computational tinkering – tools and interfaces that immerse learners in computational learning in a playful and open-ended format. We surveyed and analyzed 33 research projects to highlight the *contexts* they engage learners with their use of tangible interaction *(tangibility)* and the open-endedness or *expressivity* offered by the interfaces. Based on our survey, we develop a design space of tangible computational tinkering interfaces. Our findings (1) showcase the diversity of learning goals and creative opportunities in tinkering interfaces, (2) set forth a taxonomy of tangible interaction they utilize and (3) define a spectrum to examine the tinkerability of such interfaces. This design space provides insights for researchers, designers, and educators to explore the landscape of tangible, open-ended learning experiences and inform their future development.

Keywords and Phrases: Tangible Interaction, Tinkering, Computational Tinkering, Open-ended Learning, Design Space

1. Introduction

Tangibility has been an essential aspect of children's learning environments for centuries (e.g. Froebel's Gifts introduced in the 1840s (Brosterman, 2001)) and has been thriving in computer-supported learning environments and tools in recent decades. Resnick et al. (1996) raised the question – "instead of controlling and manipulating worlds in the computer, what if children could control and manipulate computers in the world?" "Digital manipulatives" (Zuckerman et al., 2005) have sought to engage children from the youngest of ages with computational experiences. Tangible interaction offers unique opportunities for learning with computers in ways that can be not only more intuitive and fun but also more collaborative in today's landscape of personal computing devices (Antle & Wise, 2013; Liang et al., 2021). These interfaces are often rooted in constructionist ideas – they enable children to learn by creating artifacts through the practices of making and tinkering. The tinkering mindset centers on a playful and experimental form of engagement that is driven by learners' motivation and interest (Martinez & Stager, 2013) and provides meaningful and welcoming opportunities for learners to create with computers (Vossoughi & Bevan, 2014). The term *computational tinkering* emerged as a spin on the pervasive

Ranjan, K., Mahajan, A., Vanukuru, R., & Yi-Luen Do, E. (2025). The Design Space of Tangible Interfaces for Computational Tinkerability. *Constructionism Conference Proceedings, 8/2025*, 325–340. https://doi.org/10.21240/constr/2025/102.X

narrative of *computational thinking* (Wing, 2006) to center the creative and iterative approach of tinkering – defined by Wilkinson (Presicce, 2017, p. 16) as a "playful approach to constructing with code". This term expands on the ideas of *computational thinking*, such as logic, abstraction, and analytical reasoning, to include skills like the ability to generate ideas, remix other's work, design iteratively, and being able to question and accept feedback.

In this work, we seek to develop an understanding of tangible interfaces for computational tinkering that enable learning with computational media in an open-ended, tinkering environment. We expand the definition of *computational tinkering* to not just focus on "constructing with code" but involve constructing in any computational media, based on programming and non-programming-centered computational activities such as animation, 3D modeling, etc. and diverse learning goals such as STEM learning, storytelling, etc. Past work, like Liang et al.'s review of tangible interfaces for computational tinkering (2021) and Vossoughi and Bevan's review of literature on making and tinkering (2014) have focused on the impact of tangible interfaces and tinkering experiences on children's learning respectively. Yu and Roque (2019) surveyed both commercial and research-based computational toys and kits (tangible, virtual, and hybrid) from the perspectives of design, computational learning, and expressivity. In contrast, this survey specifically examines the design of tangible interfaces that enable open-ended learning based on the tinkering mindset. We seek to develop a design space for such interfaces along the dimensions of tinkerability and tangible interaction. Design spaces, a common tool in Human-Computer Interaction (HCI), enable mapping interactive systems by articulating similarities and differences, exploring design alternatives, and ideating future possibilities (Zhang et al., 2024).

We survey past research projects and generate a design space of tangible interfaces for computational tinkering through an inductive method and purposive sampling. The research team collected a set of relevant and diverse projects to analyze – tools and interfaces that engage learners with computational experiences in an open-ended format with learner-defined rules or goals. We qualitatively analyzed our corpus of 33 papers along the dimensions of *context*, *tangibility*, and *expressivity* to answer *(RQ1) What do learners tinker with?* and *(RQ2) How do learners tinker?* For RQ1, we examine the learning goals targeted by these interfaces, their creative outputs, and the relationships between them. In RQ2, we develop a taxonomy of how TUI is utilized in such interfaces and define a "spectrum of tinkerability" offered by these projects. Our target audience for this design space is researchers, designers, and educators building and using such interfaces for learning applications. We envision this analysis to help researchers and designers better understand the characteristics and design possibilities and situate their work within the larger design space of computational tinkering.

2. Background

2.1 Tangible Interfaces for Learning

Tangible User Interfaces (TUIs) provide "tangible representations to digital information and controls, allowing users to grasp data with their hand and effect functionality by physical manipulations of these representations" (Shaer & Hornecker, 2010). TUIs offer unique opportunities for integrating digital technologies into learning environments that can be more engaging and intuitive than traditional Graphical-User

Interface (GUI) technologies. Such interfaces are generally used to support two types of learning: *exploratory* and *expressive* (Marshall, 2007) – the former engaging learners with models or representations of a topic based on existing ideas of a teacher or domain expert; the latter for learners to create external representations of their own understanding, enabling them to externalize their knowledge and reflect on it. In this review focused on tangible interfaces for computational tinkering, we will focus on TUIs for expressive activities that enable learners to tinker, develop, and represent their own ideas and knowledge.

TUIs have a range of benefits in learning environments. They offer natural and more intuitive interaction that makes them accessible to novice and young learners (Antle & Wise, 2013). These tools are often designed to be playful and engaging, and support emotional, physical, and cognitive development (Liang et al., 2021). The ease of trying different things in TUIs fosters experimentation through trial and error (Liang et al., 2021). Tangible interfaces help build a collaborative and practical learning environment for children to share ideas, objects, and their creations (Liang et al., 2021). In creative learning contexts, TUIs offer more flexibility in divergent thinking and encourage open-ended and expressive activities (Liang et al., 2021; Antle & Wise, 2013).

2.2 Tinkering and Creative Learning

The tinkering approach is a mindset involving a "playful approach to solving problems through direct experience, experimentation, and discovery" (Martinez & Stager, 2013). The open-ended and iterative style of engagement is central to the idea of constructionism – learners explore and try out ideas in the process of making something themselves (Resnick & Rosenbaum, 2013). Unlike the common "planning" mode of learning based on a top-down approach, "tinkering" recognizes a "messier" style of learning where learners react to the given context and explore different ideas. Relevant to the design of such environments, Papert (1991) especially emphasized contexts of learning in constructionism – the design of learning environments should offer opportunities for conscious engagement and reflection promoting meaningful constructionist activities instead of random trial and error.

Tinkering is an engaging, intentional, and inclusive approach to integrating STEM learning (Petrich et al., 2013). Getting "stuck" and then "unstuck" is core to tinkering – the frustration of getting stuck and then working through the problem showcases that learners are deeply engaged in the process and develop a deep understanding of materials and tools in the activity. Tinkering-based learning environments also provide meaningful opportunities to work with contextualized STEM concepts and develop making and fabrication skills such as programming, 3D printing, etc.

Tinkering approaches in learning spaces can have a range of socio-cultural benefits, as highlighted in Vossoughi and Bevan's review of making and tinkering (2014). Tinkering lowers the barrier to participation in meaningful scientific and engineering activities with its focus on play. *Epistemic Pluralism* (Turkle & Papert, 1990) in tinkering gives learners the freedom to pursue their own paths when they might otherwise be turned away by traditional STEM and engineering. This also supports children in creating their own identities as learners as they develop confidence, persistence, authorship, and new ways of thinking. Making and tinkering with technology helps learners critically rewrite their narratives from that of consumers to makers of technology. Lastly, making and tinkering activities built around communities help children learn how to collaborate, share tools and ideas, and take on new leadership and teaching roles.

3. Survey of projects

3.1 Method

Following an inductive method to develop the design space, we begin with a survey of relevant research projects. Two of the authors selected a set of papers to analyze using purposive sampling (Palinkas et al., 2015) – a standard interpretive technique involving the selection of particularly influential projects, or those that articulate interesting new areas in the space. We set three inclusion criteria for the projects in this survey: (1) involvement of tangible interaction, (2) use of a computational medium, and (3) based on the tinkering mindset. Criteria (1) focused our search on projects with a tangible interface for creation – we excluded projects where the primary interaction was through a computer or tablet, for example, Scratch (Resnick et al., 2009). With computational medium as a requirement, we filtered passive toys and unplugged activities like FlowBlocks (Threekunprapa & Yasri, 2020). For the third criterion central to the theme of the project, we focused on projects that offered expressive and playful interaction toward open-ended and meaningful outcomes. However, we found that many appropriate papers don't mention "tinkering". Therefore, we filtered for this by examining the kind of activities the papers offered and if they enabled learners to define their own rules, goals, and outcomes. We excluded works like Bots & Main(frames) (Melcer & Isbister, 2018) that present learners with various tasks to be completed.

We drew papers from a range of venues in the field of HCI and learning sciences (e.g., Interaction Design and Children; Tangible, Embedded, and Embodied Interaction; International Journal of Child-Computer Interaction; etc.) and added relevant commonly referenced papers. While some commercial toys and kits might be relevant to this survey, we limited our scope to projects with research publications in a journal or conference venue (including short papers and demos). Our final corpus for this survey was 33 papers published from 2001 to 2024. A majority of the projects were short papers (19/33) including posters, demos, and works-in-progress. 14/33 were full papers in conferences or journals. The list of all the papers included in our corpus is included in the Appendix with their short project names and labels (P*N*).

Dimension	Code	Categories
Context	Learning Goals	Programming Concepts, STEM Interest, Science Concepts, Robotics, Computational Thinking, Complex Systems, Creativity Support, etc.
	Project Output	Physical Computing, Robotics, Graphics/Art, Animation, 3D content, Data Visualization, Other Visualization, Game, Storytelling, AR, etc.
Tangibility	TUI Purpose	Represent Code, Environmental Sensing, Physical Construction, Movement Tracking, Writing and Drawing, Assemble Circuits, Represent Data
	TUI Association	Coupled, Uncoupled, Both
Expressivity	Learner-defined Rules	Yes, No
	Learner-defined Goals	Yes, No

Table 1: Coding scheme for analyzing the 33 papers in the corpus. Grouped by dimensions of context, tangibility, expressivity.

3.2 Coding and Dimensions

To analyze these projects and develop the design space, we defined a set of 6 codes to address our 3 dimensions in service to our research questions. The 6 codes and their possible categories are summarized in Table 1. To analyze the corpus of projects based on these research questions and dimensions, we followed the content analysis approach (Potter & Levine-Donnerstein, 1999) – an interpretive process where we start with a number of dimensions but no predetermined codes for each dimension, and update the code categories as we come across new project characteristics. Two authors independently coded each paper and reconciled the results to arrive at a final set of codes for each project.

1. *Dimension #1 (Context): What has the interface been designed for?* In this dimension, we looked at the Learning Goals of the projects and what the learners were creating with them (Project Output). We wanted to unveil the range of domains where tangible interfaces have been used with a tinkering mindset. These included goals like programming concepts, STEM interest, robotics, etc. while the artifact outputs created by the learners included animation, AR, visualizations, etc.
2. *Dimension #2 (Tangibility): How has tangible interaction been utilized in the project?* This dimension enabled us to consider the purpose of the tangible interface in the project (TUI Purpose) – what were the tangible elements in the interface used for? We also coded how the TUI associated with what is being created with the tool – is the TUI for creating something, or is the TUI itself the output? Projects with Uncoupled TUI provided a tangible input for a separate created artifact while projects with Coupled TUI included interfaces where the output of the creation was assimilated within the tangible itself.
3. *Dimension #3 (Expressivity): How expressive are the activities supported by the interface?* With a focus on interfaces for computational tinkering, we wanted to examine the degree of expressivity offered by these projects. Inspired by the expressivity perspective used by Yu and Roque in their review of computational toys and kits (2019), we specifically examined the design of the tools and interfaces – what kind of creations does the project enable learners to produce? Are there

predefined rules and interaction elements in the tool, or can they be experimented with and modified by the learners (Learner-defined rules)? Are the goals of the interaction predetermined or decided by learners (Learner-defined Goals)?

4. Results

Through coding and analysis of the projects in our corpus, we developed a low-level classification of each project along the dimensions of *context*, *tangibility*, and *expressivity*. Considering the relationships between these dimensions and distilling how these projects support open-ended and creative interaction helped us chart out the design space of these tangible interfaces for computational tinkering. In particular, we look at *(RQ1) what do learners tinker with?* and *(RQ2) how do learners tinker?* to develop a space for exploration of such projects. The former research question is based on the *context* dimension of the projects in our analysis, including their learning goals and output domains of the learners' creations. The latter question investigates how tangible interaction was utilized to enable tinkering and the degree of tinkerabilty these interfaces offer based on the *tangibility, expressivity,* and *context* dimensions.

4.1 What do Learners Tinker with?

Learning Goal			Project Output		
Category	Count	Example	Category	Count	Example
Programming Concepts	15/33	Quetzal & Tern [P3]	Physical Computing	10/33	littleBits [P10]
Computational Thinking	7/33	StoryBlocks [P24]	Animation	7/33	Video Puppetry [P7]
Physical Computing	7/33	Cube-In [P16]	Graphics/Art	7/33	TurTan [P5]
Computational Literacy	5/33	Draw2Code [P27]	Robotics	6/33	KIBO [P18]
STEM Interest	5/33	LilyPad [P9]	Games	6/33	T-Maze
Robotics	2/33	Molecubes [P4]	Augmented Reality (AR)	5/33	HyperCubes [P23]
3D Modeling	2/33	TADCAD [P17]	3D Content	3/33	TADCAD [P17]
Science Concepts	2/33	Topobo [P2]	Data Visualizations	2/33	SensorBricks [P33]
Complex Systems	2/33	PrototypAR [P25]	Other Visualizations	2/33	Posey [P8]
Creativity Support	2/33	Tangible Diffusion [P32]			
AR Authoring	1/33	TanCreator [P22]			

Table 2: Summary of results from the context dimension highlighting the diverse learning goals and project outputs in the corpus.

Tangible interfaces are designed to support a variety of learning goals.
The most common Learning Goal was *programming concepts* and related topics like *computational thinking* and *computational literacy*. A number of projects also sought to teach specific applications of computing such as *robotics*, *3D modeling*, and *AR authoring*. A few projects had goals not centered on computation such as *STEM* interest, *science concepts* like motion in Topobo [P2] and science modeling in Posey [P8], *complex systems*, and *creativity support*. Most projects were not limited to a single learning goal. For example, roBlocks [P6] engages learners with *computational thinking* and helps them learn about *complex systems*.

Computational tinkering supports diverse creative pursuits.
The interfaces in our corpus enabled a wide variety of Project Outputs – the types of things learners created with the tangible interface. The most common types of outputs involved physical artifacts like *physical computing* and *robotics*. *Animation* and *Graphics/Art* were also common project domains. A number of projects enabled the creation of different types of artifacts using the same interface. For instance, Kart-ON [P28] offers paper programming cards to create graphics, AR, and 3D content while Posey [P8] provides an interactive hub-and-strut model to create 3D content, animate it by interacting with the physical model, and use the model as a visualization for science concepts.

Create one thing, learn something else.
Tangible interfaces for computational tinkering were designed to support a variety of learning goals. The creative output of the projects often matched the learning outcome – such as TADCAD [P17] teaching 3D modeling principles through creation of 3D content – but they were also often used to engage with another topic. For example, there were a number of ways these projects helped teach learners about computational thinking and develop computational literacy – through robotics with KIBO [P18], with physical computing like in MakerWear [P20], and in the domain of animation and AR, Draw2Code [P27] and HyperCubes [P23]. Projects focused on teaching programming concepts did so by enabling learners to create music (e.g., Music Blocks [P26]) or graphics (e.g., TurTan [P5]). Among non-coding-centered learning goals, projects like Posey [P8] provides an interface to create animations and 3D content to visualize and understand science concepts while roBlocks [P6] uses robots as a metaphor to teach about complex systems. Video Puppetry [P7] and Tangible Diffusion [P32] aim to support learners' creativity in the domains of animation and graphics/art respectively.

4.2 How do Learners Tinker?

With this RQ, we examine how the projects in our corpus enabled learners to tinker, looking at what the TUI was used for in the project and how it was associated with the creations. This analysis describes a taxonomy of tangible interaction in learning interfaces to understand and communicate the range of tangible interaction possibilities and how tangibility might be integrated in ways that foster tinkerability.

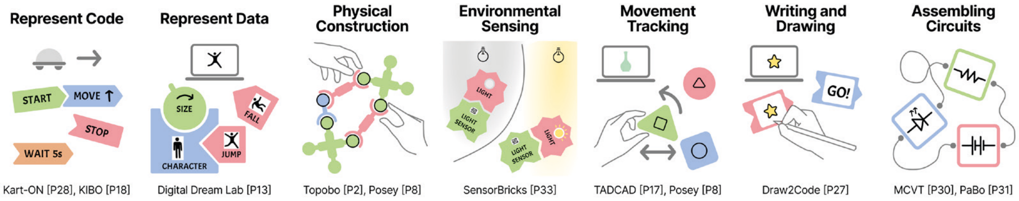


Figure 1: Taxonomy of TUI purpose in computational tinkering interfaces observed in the surveyed projects.

Through our survey, we sought to establish a taxonomy of the TUI Purpose to identify what the tangible interface was used for (Figure 1). The most common use of TUI in these projects was to *represent code* where the tangible blocks represent different programming commands or actions. These could be low-level commands to execute code functions like drawing a circle in Kart-ON [P28] or higher-level actions for a robot (e.g., KIBO [P18]) or animated characters (e.g., Roberto [P12]). Some projects use tangible blocks to *represent data* like in Digital Dream Lab [P13] to define the properties of game characters such as size, color, and animation behavior. Another common use case for the tangibles was *physical construction* where learners could build structures using the tangible blocks like in Topobo [P2] and Posey [P8]. The tangible elements in the interfaces were often used for *environmental sensing* such as in the SensorBricks [P33] toolkit with sensors for air quality, temperature, sound, etc. and for *tracking movement* of the learners' hands through the tangibles (e.g., TADCAD [P17]). A few projects involved *writing and drawing* like in Draw2Code's [P27] paper cards to draw characters to animate. Lastly, the tangible interaction in some projects was *assembling circuits* like in MCVT [P30] where learners built electrical elements and circuits on cardboard using copper tape and clips. Overall, we found 7 ways projects used TUI as illustrated in Figure 1.

Looking at the **association of the TUI** with the output, we generally find two types of project creations. *Coupled* tangible interfaces (13/33) involve projects where the project output is the tangible interface (e.g. robotic structures in Topobo [P2], wearable projects in MakerWear [P20]) or those that directly overlay the output on top of the interface using AR (e.g. HyperCubes [P23], TanCreator [P22]). On the other hand, *uncoupled* TUIs (18/33) are projects where the output is disconnected from the tangible input (e.g. visual output on a screen in Strawbies [P15], music output in Music Blocks [P26], etc.). SensorBricks [P33] and Posey [P8] are (2/33) projects that support both *coupled* and *uncoupled* TUI, offering both physical and screen-based output.

5. Discussion

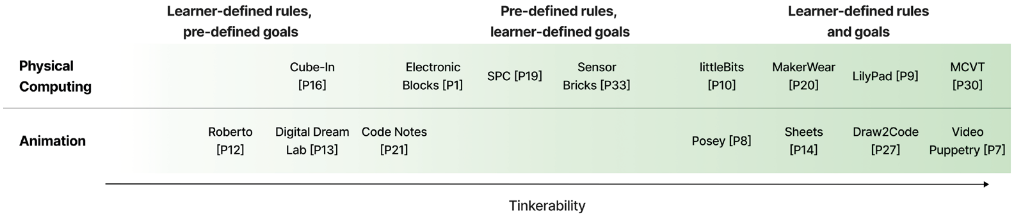
The design space presented in this work provides a framework for understanding and expanding the use of tangible interfaces in computational tinkering. By mapping out these interfaces along the dimensions of *context, tangibility* and *expressivity* to answer the questions, *"what do learners tinker with?"* and *"how do learners tinker?",* we offer insights that can inform the design of future systems.

5.1 The spectrum of tinkerability in tangible learning interfaces.

Considering the open-ended nature of the interfaces based on the *expressivity* dimension, we can begin to map out a *spectrum of tinkerability* of tangible interfaces for different domains. In our coding process, we classified each project based on whether it enabled learners to (A) define their own rules of interaction, (B) define the goal of the activity, or both. For a project to be considered an interface for tinkering, we posit that it would need to have answered yes to at least one of (A) or (B). Considering these answers and the domain of the project output, we saw a qualitative measure of tinkerability centered on the expressivity of the project. Questions (A) and (B) help determine three regions on the tinkerability spectrum in order of increasing expressivity (left-right in Figure 2): (1) learner-defined rules, pre-defined goals, (2) pre-defined rules, learner-defined goals, and (3) learner-defined rules and goals. Iterating and discussing this classification of projects, we found that the degree of expressivity can depend on the domain of the project outputs. Below, we consider two of the domains to describe how one may utilize this spectrum for analyzing and mapping other computational tinkering interfaces more generally. This spectrum maps the projects in these two domains in Figure 2 and for all the others in Figure 3 in the appendix.

In the animation domain, we consider *defining rules as* learners being able to create their own characters and define their movements freely while *defining goals* as controlling the output of the animation. Therefore, Roberto [P12] and Digital Dream Lab [P13] are placed towards the left as they both have a pre-defined character and set movements that the learners can arrange to create their own animations. Draw2Code [P27] falls on the other end of the spectrum as it enables drawing and moving your own characters. Video Puppetry [P7] is placed further right as it allows for multiple characters and free movement unlike the grid-movement in Draw2Code. In the middle of the spectrum, we have Code Notes [P21] that offers programming cards for animating preset graphics. We have 10 projects in the domain of *physical computing*, which are arranged on the spectrum as shown in Figure 3 (Appendix). The left-most is Cube-In [P16], which provides a set of blocks to engage with input-output concepts, while Electronic Blocks [P1] is to the right of it, as it supports connecting the blocks in learner-defined ways. In the middle, we have toolkits like SPC [19] and SensorBricks [P33] that support more freeform physical construction with electronic components. LittleBits [P10] and MakerWear [P20] are further right because they support integrating the creations with everyday materials like cardboard, fabrics, etc. MCVT [P30] is on the right-end of the spectrum because it engages learners to create and customize the physical computing toolkit themselves.

This design space provides a qualitative framework to assess the degree of open-endedness and expressivity offered by tangible interfaces for learning. Designers and educators can use this framework to evaluate learning experiences and identify opportunities for enhancing learner autonomy and creative agency.



	Learner-defined rules, pre-defined goals				Pre-defined rules, learner-defined goals			Learner-defined rules and goals		
Physical Computing		Cube-In [P16]		Electronic Blocks [P1]	SPC [P19]	Sensor Bricks [P33]	littleBits [P10]	MakerWear [P20]	LilyPad [P9]	MCVT [P30]
Animation	Roberto [P12]	Digital Dream Lab [P13]	Code Notes [P21]				Posey [P8]	Sheets [P14]	Draw2Code [P27]	Video Puppetry [P7]

Figure 2: The Spectrum of tinkerability illustrated for the domains of Physical Computing and Animation.

5.2 Expanding beyond programming-centered computational domains.

Since the rise of popularity of computational thinking in the last few decades, teaching programming to children through interactive tangible interfaces is seen as an effective way to develop computational thinking skills. As evidenced by our survey, there is a wealth of programming-based tangible interfaces to learn programming and develop computational literacy. However, programming is not the only domain to engage learners with computing. Traditional CS education methods centered around programming often lead youth from non-dominant groups in tech and lower socioeconomic status to be discouraged and lose interest in STEM and computing. Therefore, it is important to support a diverse range of computational activities that cater to different learner interests.

In our survey, we observed how tangible interfaces for computational tinkering can support learning across a variety of non-programming computational domains like animation and 3D modeling as well as typically non-computational domains like storytelling, science learning, music, etc. These projects also engage learners through more embodied and expressive interaction modalities such as movement tracking, writing and drawing, environmental sensing. We advocate for tangible learning activities that are valuable and welcoming for learners of all backgrounds and that promote learners' autonomy in developing their own interests and identities in today's technology landscape. By broadening the scope of activities, designers can create more inclusive and diverse tinkering experiences that cater to different learning styles, interests, and disciplines.

6. Conclusion

In this paper, we present our literature survey and design space for computational tinkerability with tangible tools and interfaces, exploring previous projects that engage learners in creative computational experiences. We investigated and analyzed 33 tangible tools and interfaces along the dimensions of *context*, *tangibility*, and *expressivity*. To answer *(RQ1) what do learners tinker with?*, our survey revealed a variety of learning goals and project outputs, and in particular, how diverse project outputs can be used to support different learning goals. For our question, *(RQ2) how do learners tinker?*, we synthesized a taxonomy of tangible interaction in tinkering interfaces and a spectrum of tinkerability based on a tangible interface's support of learner-defined rules and goals. We hope our design space furthers understanding and inspires future work expanding the use of tangible interfaces in computational tinkering.

Acknowledgements

This research is sponsored in part by the U.S. National Science Foundation through grant IIS-2040489.

References

Antle, A. N., & Wise, A. F. (2013). Getting down to details: Using theories of cognition and learning to inform tangible user interface design. *Interacting with Computers*, 25(1), 1-20.

Barnes, C., Jacobs, D. E., Sanders, J., Goldman, D. B., Rusinkiewicz, S., Finkelstein, A., & Agrawala, M. (2008). Video puppetry: a performative interface for cutout animation. In *ACM SIGGRAPH Asia 2008 papers* (pp. 1-9).

Bdeir, A. (2009, February). Electronics as material: littleBits. In Proceedings of the 3rd international conference on tangible and embedded interaction (pp. 397-400).

Brombacher, H., Van Koningsbruggen, R., Vos, S., & Houben, S. (2024, February). SensorBricks: a Collaborative Tangible Sensor Toolkit to Support the Development of Data Literacy. In *Proceedings of the Eighteenth International Conference on Tangible, Embedded, and Embodied Interaction* (pp. 1-17).

Brosterman, N. (2001). Inventing kindergarten. *Journal of Aesthetics and Art Criticism*, 59(2).

Buechley, L., Eisenberg, M., Catchen, J., & Crockett, A. (2008, April). The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 423-432).

Cassinelli, A., & Saakes, D. (2017, March). Data flow, spatial physical computing. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction* (pp. 253-259).

Fuste, A., & Schmandt, C. (2019, October). HyperCubes: A playful introduction to computational thinking in augmented reality. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts* (pp. 379-387).

Gallardo, D., Julia, C. F., & Jorda, S. (2008, October). TurTan: A tangible programming language for creative exploration. In *2008 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems* (pp. 89-92). IEEE.

Han, K., Tang, K., & Wang, M. (2024, February). Tangible Diffusion: Exploring Artwork Generation via Tangible Elements and AI Generative Models in Arts and Design Education. In *Proceedings of the Eighteenth International Conference on Tangible, Embedded, and Embodied Interaction* (pp. 1-13).

Horn, M. S., AlSulaiman, S., & Koh, J. (2013, June). Translating Roberto to Omar: computational literacy, stickerbooks, and cultural forms. In *Proceedings of the 12th International Conference on Interaction Design and Children* (pp. 120-127).

Horn, M. S., & Jacob, R. J. (2007, February). Designing tangible programming languages for classroom use. In *Proceedings of the 1st international conference on Tangible and embedded interaction* (pp. 159-162).

Hu, F., Zekelman, A., Horn, M., & Judd, F. (2015, June). Strawbies: explorations in tangible programming. In *Proceedings of the 14th International Conference on Interaction Design and Children* (pp. 410-413).

Im, H., & Rogers, C. (2021, June). Draw2code: Low-cost tangible programming for creating AR animations. In *Proceedings of the 20th Annual ACM Interaction Design and Children Conference* (pp. 427-432).

Jin, Q., Wang, D., & Sun, F. (2018, October). TanCreator: A Tangible Tool for Children to Create Augmented Reality Games. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers* (pp. 82-85).

Kang, S., Norooz, L., Bonsignore, E., Byrne, V., Clegg, T., & Froehlich, J. E. (2019, June). Prototypar: Prototyping and simulating complex systems with paper craft and augmented reality. In *Proceedings of the 18th ACM international conference on interaction design and children* (pp. 253-266).

Kazemitabaar, M., McPeak, J., Jiao, A., He, L., Outing, T., & Froehlich, J. E. (2017, May). Makerwear: A tangible approach to interactive wearable creation for children. In *Proceedings of the 2017 chi conference on human factors in computing systems* (pp. 133-145).

Koushik, V., Guinness, D., & Kane, S. K. (2019, May). Storyblocks: A tangible programming game to create accessible audio stories. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (pp. 1-12).

Liang, M., Li, Y., Weber, T., & Hussmann, H. (2021, June). Tangible interaction for children's creative learning: A review. In *Proceedings of the 13th Conference on Creativity and Cognition* (pp. 1-14).

Long, D., Rollins, S., Ali-Diaz, J., Hancock, K., Nuonsinoeun, S., Roberts, J., & Magerko, B. (2023, June). Fostering AI Literacy with Embodiment & Creativity: From Activity Boxes to Museum Exhibits. In *Proceedings of the 22nd Annual ACM Interaction Design and Children Conference* (pp. 727-731).

Marshall, P. (2007, February). Do tangible interfaces enhance learning?. In *Proceedings of the 1st international conference on Tangible and embedded interaction* (pp. 163-170).

Martinez, S. L., & Stager, G. S. (2013). Invent to learn: Makers in the classroom. *The Education Digest*, 79(4), 11.

Melcer, E. F., & Isbister, K. (2018, April). Bots & (Main) frames: exploring the impact of tangible blocks and collaborative play in an educational programming game. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (pp. 1-14).

Oh, H., Deshmane, A., Li, F., Han, J. Y., Stewart, M., Tsai, M., ... & Oakley, I. (2013, February). The digital dream lab: tabletop puzzle blocks for exploring programmatic concepts. In *Proceedings of the 7th international conference on tangible, embedded and embodied interaction* (pp. 51-56).

Oh, H., & Gross, M. D. (2015, January). Cube-in: A learning kit for physical computing basics. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction* (pp. 383-386).

Oh, H., Hsi, S., Posner, N., Dixon, C., Smith, T., & Cheng, T. (2023). Making computing visible & tangible: A paper-based computing toolkit for codesigning inclusive computing education activities. *International Journal of Child-Computer Interaction*, *38*, 100602.

Palinkas, L. A., Horwitz, S. M., Green, C. A., Wisdom, J. P., Duan, N., & Hoagwood, K. (2015). Purposeful sampling for qualitative data collection and analysis in mixed method implementation research. *Administration and policy in mental health and mental health services research*, 42, 533-544.

Papert, S. (1991). Situating Constructionism. *Constructionism/Ablex*.

Petrich, M., Wilkinson, K., & Bevan, B. (2013). It looks like fun, but are they learning?. In *Design, make, play* (pp. 50-70). Routledge.

Potter, W. J., & LevineDonnerstein, D. (1999). Rethinking validity and reliability in content analysis.

Presicce, C. (2017). Explorations in computational tinkering (Doctoral dissertation, Massachusetts Institute of Technology).

Raffle, H. S., Parkes, A. J., & Ishii, H. (2004, April). Topobo: a constructive assembly system with kinetic memory. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 647-654).

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67.

Resnick, M., & Rosenbaum, E. (2013). Designing for tinkerability. In Design, make, play (pp. 163-181). Routledge.

Sabuncuoglu, A. (2020, February). Tangible music programming blocks for visually impaired children. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* (pp. 423-429).

Sabuncuoğlu, A., Erkaya, M., Buruk, O. T., & Göksun, T. (2018, June). Code notes: designing a low-cost tangible coding tool for/with children. In *Proceedings of the 17th ACM Conference on Interaction Design and Children* (pp. 644-649).

Sabuncuoglu, A., & Sezgin, T. M. (2022). Kart-on: An extensible paper programming strategy for affordable early programming education. *Proceedings of the ACM on Human-Computer Interaction*, *6*(EICS), 1-18.

Schweikardt, E., & Gross, M. D. (2008, November). Learning about complexity with modular robots. In *2008 Second IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning* (pp. 116-123). IEEE.

Shaer, O., & Hornecker, E. (2010). Tangible user interfaces: past, present, and future directions. *Foundations and Trends® in Human–Computer Interaction*, 3(1–2), 4-137.

Sullivan, A. A., Bers, M. U., & Mihm, C. (2017). Imagining, playing, and coding with KIBO: using robotics to foster computational thinking in young children. *Siu-cheung KONG The Education University of Hong Kong, Hong Kong, 110.*

Tada, K., & Tanaka, J. (2015). Tangible programming environment using paper cards as command objects. *Procedia Manufacturing, 3*, 5482-5489.

Te, P. (2015, June). TADCAD: A tangible and gestural 3D modeling & printing platform for building creativity. In *Proceedings of the 14th International Conference on Interaction Design and Children* (pp. 406-409).

Threekunprapa, A., & Yasri, P. (2020). Unplugged Coding Using Flowblocks for Promoting Computational Thinking and Programming among Secondary School Students. *International Journal of Instruction*, 13(3), 207-222.

Turkle, S., & Papert, S. (1990). Epistemological pluralism: Styles and voices within the computer culture. *Signs: Journal of women in culture and society*, 16(1), 128-157.

Vossoughi, S., & Bevan, B. (2014). Making and tinkering: A review of the literature. National Research Council Committee on Out of School Time STEM, 67, 1-55.

Wang, D., Zhang, C., & Wang, H. (2011, June). T-Maze: a tangible programming tool for children. In *Proceedings of the 10th international conference on interaction design and children* (pp. 127-135).

Weller, M. P., Do, E. Y. L., & Gross, M. D. (2008, February). Posey: instrumenting a poseable hub and strut construction toy. In *Proceedings of the 2nd international conference on Tangible and embedded interaction* (pp. 39-46).

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

Wyeth, P., & Wyeth, G. F. (2001). Electronic blocks: Tangible programming elements for preschoolers.

Yang, R., & Do, E. Y. L. (2024, March). PaBo Bot: Paper Box Robots for Everyone. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction* (pp. 1158-1162).

Yu, J., & Roque, R. (2019). A review of computational toys and kits for young children. *International Journal of Child-Computer Interaction*, 21, 17-36.

Zhang, A. X., Bernstein, M. S., Karger, D. R., & Ackerman, M. S. (2024). Form-From: A Design Space of Social Media Systems. *Proceedings of the ACM on Human-Computer Interaction*, 8(CSCW1), 1-47.

Zuckerman, O., Arida, S., & Resnick, M. (2005, April). Extending tangible interfaces for education: digital montessori-inspired manipulatives. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 859-868).

Zykov, V., Chan, A., & Lipson, H. (2007, November). Molecubes: An open-source modular robotics kit. In *IROS-2007 Self-Reconfigurable Robotics Workshop* (pp. 3-6). Citeseer.

Appendix

Label	Title	Author	Year	Short name
P1	Electronic Blocks: Tangible Programming Elements for Preschoolers	Wyeth & Wyeth	2001	Electronic Blocks
P2	Topobo: A Constructive Assembly System with Kinetic Memory	Raffle et al.	2004	Topobo
P3	Designing tangible programming languages for classroom use	Horn & Jacob	2007	Quetzal & Tern
P4	Molecubes: An Open-Source Modular Robotics Kit	Zykov et al.	2007	Molecubes
P5	TurTan: A Tangible Programming Language for Creative Exploration	Gallardo et al.	2008	TurTan
P6	Learning about Complexity with Modular Robots	Schweikardt & Gross	2008	roBlocks
P7	Video Puppetry: A Performative Interface for Cutout Animation	Barnes et al.	2008	Video Puppetry
P8	Posey: Instrumenting a Poseable Hub and Strut Construction Toy	Weller et al.	2008	Posey
P9	The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education	Buechley et al.	2008	LilyPad
P10	Electronics as material: littleBits	Bdeir	2009	littleBits
P11	T-Maze: A Tangible Programming Tool for Children	Wang et al.	2011	T-Maze
P12	Translating Roberto to Omar: Computational Literacy, Stickerbooks, and Cultural Forms	Horn et al.	2013	Roberto
P13	The Digital Dream Lab: Tabletop Puzzle Blocks for Exploring Programmatic Concepts	Oh et al.	2013	Digital Dream Lab
P14	Tangible Programming Environments using Paper Cards as Command Objects	Tada & Tanaka	2015	Sheets
P15	Strawbies: Explorations in Tangible Programming	Hu et al.	2015	Strawbies
P16	Cube-in: A Learning Kit for Physical Computing Basics	Oh & Gross	2015	Cube-in
P17	TADCAD: a tangible and gestural 3D modeling & printing platform for building creativity	Te	2015	TADCAD
P18	Imagining, Playing, and Coding with KIBO: Using Robotics to Foster Computational Thinking in Young Children	Sullivan et al.	2017	KIBO
P19	Data Flow, Spatial Physical Computing	Cassinelli & Saakes	2017	SPC
P20	MakerWear: A Tangible Approach to Interactive Wearable Creation for Children	Kazemitabaar et al.	2017	MakerWear

Label	Title	Author	Year	Short name
P21	Code Notes: Designing a Low-Cost Tangible Coding Tool for/with Children	Sabuncuoglu et al.	2018	Code Notes
P22	TanCreator: A Tangible Tool for Children to Create Augmented Reality Games	Jin et al.	2018	TanCreator
P23	HyperCubes: A Playful Introduction to Computational Thinking in Augmented Reality	Fuste & Schmandt	2019	HyperCubes
P24	StoryBlocks: A Tangible Programming Game to Create Accessible Audio Stories	Koushik et al.	2019	StoryBlocks
P25	PrototypAR: Prototyping and Simulating Complex Systems with Paper Craft and Augmented Reality	Kang et al.	2019	PrototypAR
P26	Tangible Music Programming Blocks for Visually Impaired Children	Sabuncuoglu	2020	Music Blocks
P27	Draw2Code: Low-Cost Tangible Programming for Creating AR Animations	Im & Rogers	2021	Draw2Code
P28	Kart-ON: An Extensible Paper Programming Strategy for Affordable Early Programming Education	Sabuncuoglu & Sezgin	2022	Kart-ON
P29	Fostering AI Literacy with Embodiment and Creativity	Long et al.	2023	Knowledge Net & Creature Features
P30	Making computing visible & tangible: A paper-based computing toolkit for codesigning inclusive computing education activities	Oh et al.	2023	MCVT
P31	PaBo Bot: Paper Box Robots for Everyone	Yang & Do	2024	PaBo Bot
P32	Tangible Diffusion: Exploring Artwork Generation via Tangible Elements and AI Generative Models in Arts and Design Education	Han et al.	2024	Tangible Diffusion
P33	SensorBricks: A Collaborative Tangible Sensor Toolkit to Support the Development of Data Literacy	Brombacher et al.	2024	SensorBricks

Table 3: Final corpus of 33 papers.

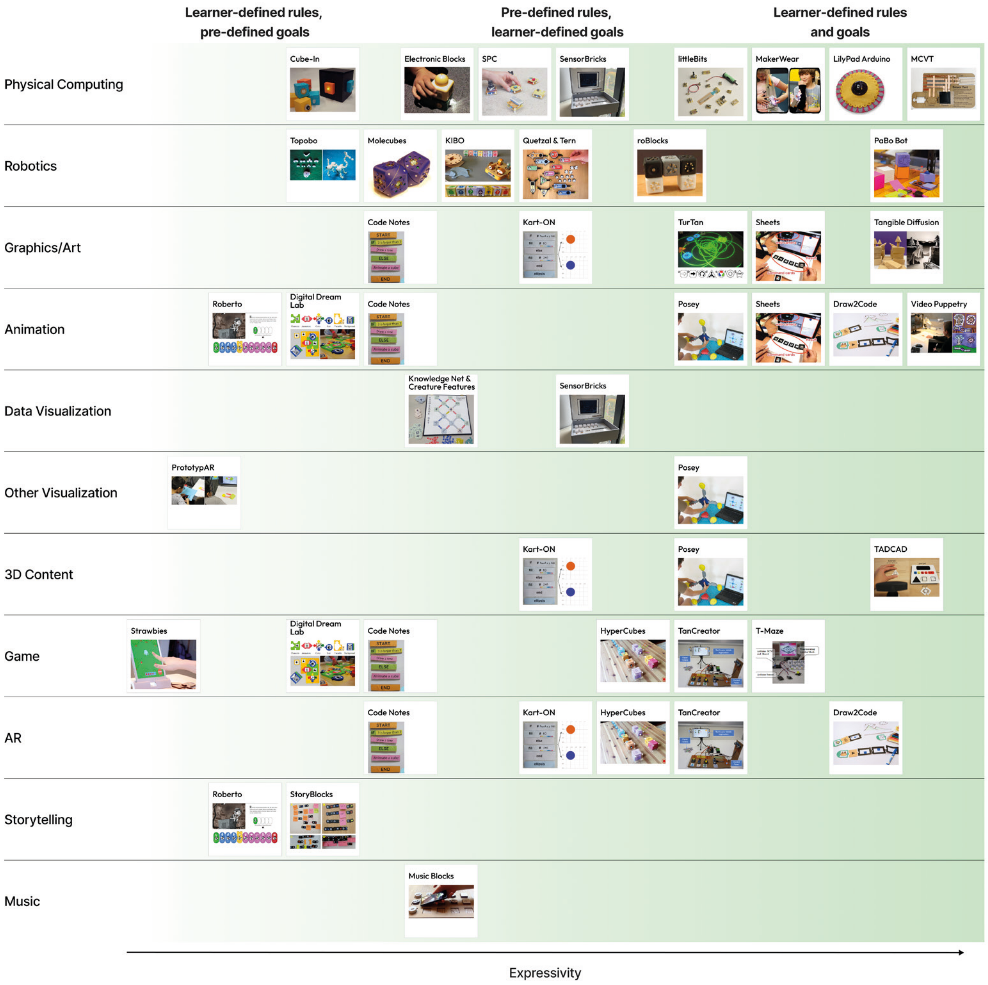


Figure 3: The Spectrum of Tinkerability illustrated for all the projects and their domains.