Designing Looms as Kits for Collaborative Assembly

Samantha Speer
Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
snspeer@andrew.cmu.edu

Nickolina Yankova
UCI
Irvine, California, USA
nyankova@uci.edu

Joey Huang
North Carolina State University
Houston, North Carolina, USA
joeyhuang@ncsu.edu

Carolyn Rosé
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
cprose@cs.cmu.edu

Kylie A Peppler
School of Education / Creativity Labs
University of California, Irvine
Irvine, California, USA
kpeppler@gmail.com

James McCann
Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
jmccann@cs.cmu.edu

Melisa Orta Martinez
Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA
mortamar@andrew.cmu.edu

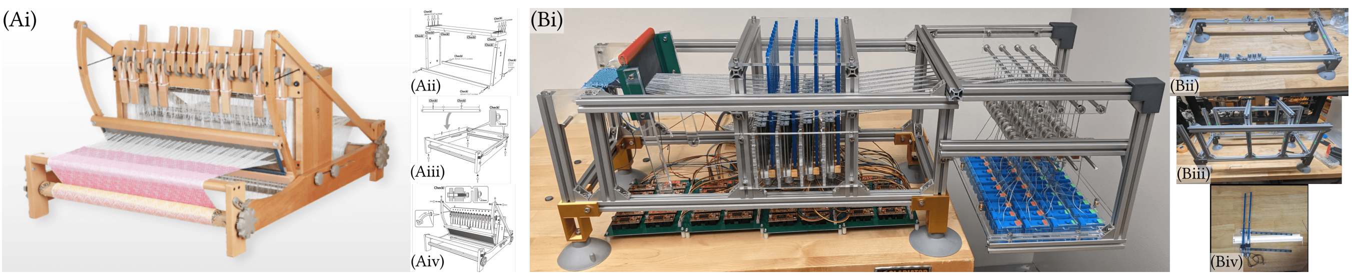


Figure 1: The Ashford Loom [48] (Ai) and RoboLoom [40] (Bi) in their built and partially built stages. For the Ashford loom, (Aii) shows the castle assembly, (Aiii) shows the frame assembly, and (Aiv) shows the beater assembly. For RoboLoom, (Bii) shows the base frame assembled, (Biii) shows the final frame assembled, and (Biv) shows partial assembly of the motors.

Abstract

Both engineering kits and collaboration skills are increasingly prevalent in education and are often used in conjunction, yet little research addresses how kit hardware can be designed to support collaboration. Collaboration skills are best acquired through carefully designed collaborative experiences. While prior work highlights technology's important role in fostering collaboration, less research explicitly explores hardware design. This paper posits that certain design features of kits can promote or hinder collaboration. We present a user study evaluating the collaborative assembly of two loom kits: one for higher education (RoboLoom) and one for individuals (Ashford Loom). We developed a coding scheme from the 3Cs framework (coordination, cooperation, and communication) to analyze which hardware features influenced collaboration. We find five design feature categories that may influence collaboration: repetitiveness, specificity, difficulty, parallelizability, and physicality. This paper presents our findings and recommendations for implementing these features into educational kit hardware design to create opportunities for collaboration.

This work is licensed under a Creative Commons Attribution 4.0 International License.

UIST '25, Busan, Republic of Korea

© 2025 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-2037-6/25/09
https://doi.org/10.1145/3746059.3747765

CCS Concepts

• **Hardware** → *Emerging interfaces*; • **Human-centered computing** → *Interaction devices*; **Collaborative and social computing devices**; **Empirical studies in interaction design**.

Keywords

Collaboration, Hardware Design, Engineering Kits

ACM Reference Format:
Samantha Speer, Nickolina Yankova, Joey Huang, Carolyn Rosé, Kylie A Peppler, James McCann, and Melisa Orta Martinez. 2025. Designing Looms as Kits for Collaborative Assembly. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25), September 28–October 01, 2025, Busan, Republic of Korea.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3746059.3747765

1 Introduction

Assemblable engineering kits have become prevalent in engineering education as they provide suites of building materials, sensors, and other hardware components for students to assemble and learn with. Most research on the use and design of engineering kits focuses on individual positive learning outcomes, despite these kits often being used in groups [34, 38]. While most studies report the kits being used in groups, with some work even showing increases in collaboration [12], there is little research on designing the hardware specifically to support collaborative experiences. Viewing educational kits through the lens of user interface design prompts us to ask which features of a kit's hardware and assembly process (effectively, its "tangible user interface") afford a collaborative user experience.

Collaborative experiences help build essential skills that prepare students for their future engineering careers in academia or industry [1–3]. Collaboration skills can be fostered through the use of collaborative experiences during learning tasks [24, 47]. However, for this to occur, the collaborative experiences must be carefully designed to provide scaffolding for collaborative skills [9]. The field of Computer-Supported Collaborative Learning focuses on supporting collaboration during learning through the use of information and communication technologies [45]. Research shows that software-supported collaborative learning can be beneficial for engineering education [18, 26, 29, 44]. However, there is little research on how engineering educational hardware kits can be designed to support collaborative actions. Research shows that challenges (such as poor task design) can prevent collaborative actions and hinder learning outcomes [18, 26]. To support collaborative actions, all aspects of collaborative learning tasks, including the hardware, should be designed to allow for and support collaborative actions [26].

In this paper, we present a crossover study where two groups of participants (n=8) built two functionally similar loom kits designed for different audiences: RoboLoom [40] and the Ashford Loom [48] (Figure 1). In their work, the authors state that RoboLoom is designed to be assembled in groups to support interdisciplinary collaborative learning [40]. The Ashford Loom is commercially available and designed to be assembled by individuals for personal weaving [48]. We chose these devices because they are designed for different audiences, but have similar end uses (weaving). By examining differences in the assembly process of these two loom kits, we aim to gain insights into the influences of hardware design features on collaborative behaviors.

To analyze participants' collaborative behaviors during building, we developed and applied a coding scheme based on the 3C's model of collaboration [14, 17] (Coordination, Cooperation, Communication). We then used thematic analysis of these actions within codes to find hardware features that promoted and hindered collaboration. Particularly, we found five categories of hardware features that created the opportunity for collaborative actions: repetitiveness (DP1), specificity (DP2), difficulty (DP3), parallelizability (DP4), and physicality (DP5). We found that collaboration was influenced by these hardware principles in the following ways:

- C1. Coordination can be facilitated by hardware that is repetitive (DP1), semi-specific (DP2), physically and mentally challenging (DP3), parallelizable (DP4), and requires simultaneous manipulation (DP5).
- C2. Cooperation is possible with hardware that is repetitive (DP1), physically challenging (DP3), parallelizable (DP4), or requires simultaneous manipulation (DP5).
- C3. Communication is encouraged by hardware that is repetitive (DP1), is semi-specific (DP2), is physically and mentally challenging (DP3), or requires simultaneous manipulation (DP5).
- C4. Collaboration can be blocked by hardware that is too mentally challenging (DP3), not parallel (DP4), is small (DP5), or does not require simultaneous manipulation (DP5).

We hope that this paper begins a discussion of designing hardware for collaborative interactions, even if these deviate from "good design" principles such as ease or speed of assembly. Since many educational kits are student-assembled and used collaboratively to save cost, we argue for leveraging these trade-offs as an opportunity for collaborative learning. Our results highlight trade-offs between reducing assembly difficulty and using that difficulty to foster collaboration.

2 Related Work

In this section, we discuss the definition of collaboration we use and the 3Cs framework [14, 17] for further categorizing collaborative actions to understand more specific behaviors. We additionally discuss how technology has been designed to support collaborative actions in the past and how educational kits have been studied.

2.1 Collaboration and Computer-Supported Cooperaton

In this paper, we will define collaboration as a process of two or more people collectively working towards a shared goal where the output of the group cannot be easily separated into individual contributions [10, 13, 28, 36, 46]. We draw this definition from past theoretical frameworks for defining collaboration from the vast body of work in Computer-Supported Collaborative Learning (CSCL) and Computer-Supported Cooperative Work (CSCW). We also adopt the 3Cs model (Coordination, Cooperation, and Communication) proposed by Ellis et al. [14, 17] for further specifying collaborative actions. We define coordination as the process of organizing the people, activities, and resources necessary to accomplish a shared goal, ensuring shared understanding about the state of each [11, 14, 17, 28, 36]. Conversely, cooperation is the process of individuals working on achieving specific tasks that contribute to a shared goal [5, 17, 28, 36]. Finally, we define communication as the process of negotiating meaning across contexts relevant to the shared task [14, 17, 28].

Based on this research, technology that supports collaboration must be designed for supporting coordination, cooperation, and communication [28] and thus designed for the support of shared meaning across different contexts, of organization and shared understanding of state space between actors, and of joint operations in the workspace [28]. Designing effective collaborative problems requires tasks that make working together necessary for the achievement of the task, and that goals cannot be achieved by the individual [5]. Kirschner et al. [21] provide further guidelines for the design of educational collaborative technologies, including technological usability and educational support, stating that the technology should fulfill the learning intentions of the group instantaneously and guide members to use the learning interventions when needed. They also state that social affordances should be considered in designing a CSCL environment to facilitate relevant social interactions for learners [22]. These guidelines provide broad design principles for consideration when designing general systems. In this paper, we take these broad guidelines and apply them to hardware, specifically, assembleable looms for classroom use.

Towards these goals, there have been many technologies developed to support collaboration in different contexts [4, 12, 19, 20, 23, 25, 28, 35, 39]. Much research has been done into how mixed reality applications can support collaboration through shared information [19, 25, 39, 42]. Conversely, Knoll et al. [23] showed that strategically hiding information led to more communication between players of a mixed reality escape room game. The co-location aspect of collaborative mixed reality applications has also been explored, showing that co-location leads to more collaboration [23, 35]. Additionally, the use of hardware has been shown to increase the feeling of co-location during remote collaboration tasks [20]. In addition to co-location, the physicality of collaborative hardware has been found to affect collaboration as well [4, 12, 28, 42]. Africano et al. [4] designed a tangible interface for collaborative classroom interactions, finding that the large scale of the device allowed multiple children to use the device simultaneously. Moura et al. [28] find similar results with their interactive tangible device, adding the result that multiple simultaneous physical inputs also allow for more collaboration. In work by Stanton et al. [42], the authors hypothesize that the large-scale input device and use of physical props as input allow for more collaboration as users are slower to act, leaving time for others to contribute. This initial work provides evidence that the physicality of devices can affect collaboration. In this paper, we explore further how hardware design can affect collaboration beyond its existence and scale.

2.2 Educational Kits

Educational kits have recently become popular for robotics and engineering learning [7, 34]. Educational kits are examples of tangible user interfaces [7], and many of the design principles focus on the tangibility of the kits to support constructivist learning [38]. Much research has gone into the hardware design of these kits in their final forms to encourage specific interactions, showing them to be effective for individual learning outcomes when teaching across a range of ages [8, 15, 16, 27, 34, 37, 43]. Due to cost and other pedagogical choices, these kits are often used in groups [8, 15, 16, 27, 34, 37, 43]. Though kits are used in groups, research focuses on the students' individual learning gains within these groups, leaving the effectiveness of collaborative actions within the groups largely unexplored. This presents an opportunity to study the effect of the hardware design on group interactions to add additional potential gains to the preestablished pedagogical routine of implementing educational hardware kits in group settings. Furthermore, there are many different kits that range in educational topics, programmability, and configurability [34], but most require assembly by users. Most research on user interface design for single-purpose end products focuses on the final state of the hardware [7], leaving the opportunity to explore the disassembled hardware as an interface of its own, eliciting a different set of interactions than the final use state of the kit. In this paper, we aim to explore this opportunity by examining group interactions with hardware from educational [40] and non-education-focused [48] kits to understand how the design of the hardware can impact collaboration. We chose to use looms as an initial starting point for studying hardware design features of kits, as there are decades of design experience in looms for end-user assembly, as well as the more recent trend of designing do-it-yourself loom kits for higher education engineering instruction [40].

3 Methods

In this exploratory study, participants collaborated in groups to construct two hardware kits: RoboLoom and the Ashford Loom. The study aims to gain insight into how hardware design features and the interactions they afford can support or hinder collaborative actions during the assembly process.

3.1 Weaving Machine Kits

We chose to use looms as an initial starting point for studying hardware design features that can eventually be studied more generally. This choice was made due to the existence of decades of design for looms for end-user assembly, as well as the existence of do-it-yourself kits that have begun to be used for higher education engineering instruction [40].

We chose RoboLoom due to the prior research showing initial differing interactions with assembling the RoboLoom [40]. We chose the Ashford Loom (Figure 1) because it is a commercial counterpart designed for end-user assembly. As they are both looms, they share common design features for loom functionality, but differ in the specific design and assembly of these features. For example, both looms have a base frame that holds the mechanism for yarn manipulation in the middle. More details on loom design and assembly are given in Appendix A. We use this as an opportunity to observe collaborative actions with hardware designed in different ways for similar functionality.

These kits are designed for different purposes, with RoboLoom being specifically designed for educational applications and the Ashford Loom for individual hobbyist use. As such, the design choices yield different features. For example, the Ashford Loom uses Philips head screws and hardwood to build its frame (Appendix A, Figure 17) as these are common and recognizable to many end users. RoboLoom uses machine screws and common aluminum framing corner brackets to create its frame (Appendix A, Figure 14) as these are typical of engineering applications and provide an opportunity to learn about engineering practices [40]. We use this as an opportunity to observe collaborative actions across a broader range of hardware.

3.2 Participants

Eight students participated in the study, recruited via email lists, Slack, etc., including informatics graduate students. Seven were graduate students in Informatics or Education, and one was an

Table 1: Time spent on assembly per group per loom given in the format of h:mm:ss as well as the number of data points (5-minute segments) in the data log.

Group	RoboLoom		Ashford Loom	
	Time	Data Points	Time	Data Points
1	5:47:12	70	4:36:20	56
2	5:02:36	61	2:54:12	35

undergraduate student in Computer Science. Participants were grouped into two groups of four and distributed so that each group had a diverse disciplinary background. All participants had diverse sets of expertise in the arts, design, science, technology, engineering, and mathematics (STEM), programming, and human-computer interaction (HCI). Both groups engaged in the construction of both looms – Group 1 built RoboLoom first, then the Ashford Loom, while Group 2 did the reverse. Assembly sessions were ≈3-6 hours per loom (Table 1) over four days. Participants were provided with instruction manuals (Appendix A) and the necessary tools to complete the assembly. Facilitators were available to answer questions along the way, without intervening in group work. This study focuses on investigating the hardware features that support or hinder group collaboration. The aim of the analysis is exploratory, rather than to determine causation.

3.3 Data Sources and Analysis

We collected audio and video data from the assembly sessions and constructed one data log per group per loom (four data logs total). Each log documented the assembly process in increments of 5 minutes. Each 5-minute section was summarized and labeled with the assembly step(s), engaged hardware, and group members. Table 1 shows the number of 5-minute segments (Data points) as well as the total assembly time for each loom. Figure 2 gives a graphical summary of the coded data.

In our subsequent iterative analysis, we selected the joint collaborative actions as the unit of analysis [41] to understand the actions taken by the group as a whole, rather than examining individual contributions. We developed our coding scheme by building on existing observational data of students building the RoboLoom in a prior classroom study [40], which revealed different group dynamics during and after assembly. Cooperative hardware interaction on different steps may have led to disjointed group dynamics. Cooperation on the same hardware may have led to more group cohesion.

We then combined these emergent observations with the 3Cs model of collaboration to specifically capture the nuance between the modes of collaboration (coordination, cooperation, and communication). From this, we defined four code categories with specific codes in each grounded in both data and theory. Example codes are shown in Appendix B. These codes are not mutually exclusive and often appear together as coordination, cooperation, and communication build together to create collaboration as a whole. Additionally, we code for instances where collaboration was attempted, but not completeable due to hardware design. This blocked collaboration could lead to adverse effects, like unwillingness to participate or negative affect [30, 32]. The codes are:

(1) *Coordination* - participants are engaging in behavior to manage others, activities, or resources needed for assembly.
  (a) *Coordination of People* - Participants are managing the distribution of labor.
  (b) *Coordination of Materials* - Participants are managing the materials for the task.
  (c) *Coordination of Specific Tasks* - Participants are managing how and when to accomplish their activities.
(2) *Cooperation* - participants work separately on tasks that both contribute to the shared assembly goal.
  (a) *Cooperation - Different Steps* - Participants or participant groups work on two steps of assembly concurrently.
  (b) *Cooperation - Same Step, Different Hardware* - Participants are working on the same step of assembly, but interacting with distinct hardware pieces.
  (c) *Cooperation - Same Step, Same Hardware* - Participants are working on the same step of assembly and interacting with the same loom materials.
(3) *Communication* - Participants discuss to share, create, and clarify knowledge of assembly.
  (a) *Helping: Knowledge Co-Creation* - Participants are creating knowledge about the loom's assembly with each other. This requires contributions from all participants involved to build the knowledge together.
  (b) *Helping: Knowledge Transfer* - One participant transfers knowledge to another, intending to aid the second in the assembly task.
  (c) *Communicating About Troubleshooting* - Participants talk through troubleshooting when there is an error in the assembly process.
  (d) *Communicating Instructions Confusion* - Participants communicate when instructions are unclear.
(4) *Blocked Collaboration* - A participant attempted collaboration but did not succeed. This does not include participants who are off task.

For each five minutes of assembly, we coded joint actions with the applicable code(s) and the engaged design feature. Two coders coded the data, with 20% of the data coded by both to determine Cohen's Kappa for inter-rater reliability (Table 2). As shown in Figure 2, the results from our coding show coordination, cooperation, and communication co-existing within the same 5 minutes. This is expected as all three are often a part of a successful collaborative action. We then performed thematic analysis for each code to understand what hardware features supported or hindered collaboration through cooperation, coordination, and communication.

4 Findings

We qualitatively analyzed the observation data to determine the percentage of time groups spent collaborating and which design features they interacted with while collaborating. We summarize our findings in Figures 3, 4, 5, and 6, which report the percentage of data points (5-minute segments) that received each code. Percentages were calculated by counting the number of data points marked with that code, and then dividing by the total number of data points (found in Table 1).

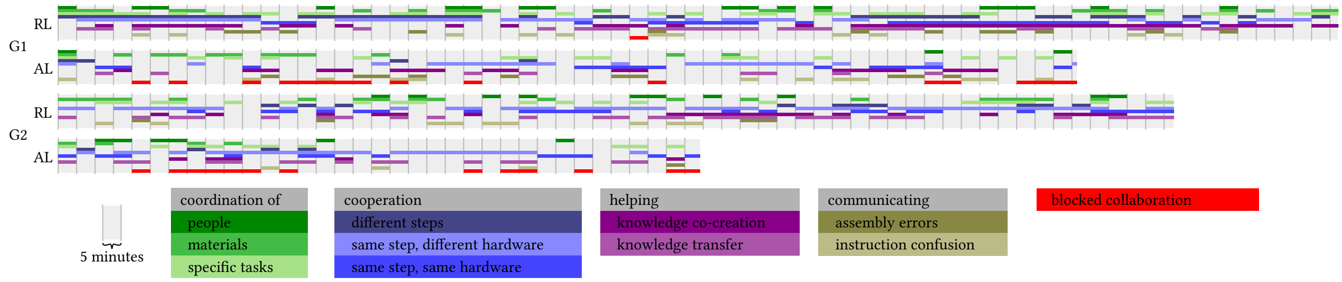


Figure 2: A graphical summary of the codes assigned to each 5-minute block of each loom construction session. "RL" and "AL" denote the RoboLoom and Ashford Loom, respectively.

Table 2: Inter-rater Reliability for each code.

Code	Cohen's Kappa (κ)
Coordination of People	0.88
Coordination of Materials	1.00
Coordination of Specific Tasks	0.91
Coop - Different Steps	1.00
Coop - Same Step, Different Hardware	1.00
Coop - Same Step, Same Hardware	1.00
Com - Helping: Knowledge Co-Creation	0.88
Com - Helping: Knowledge Transfer	0.94
Communicating About Troubleshooting	0.85
Communicating Instruction Confusion	0.93
Blocked Collaboration	1.00

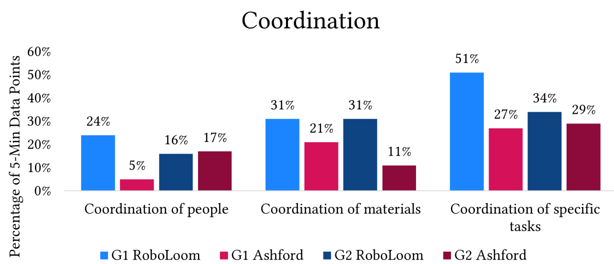


Figure 3: The percentage of 5-minute data points coded for the coordination codes for each of the four test cases. The bars show higher coordination across codes in at least one RoboLoom build.

4.1 Coordination

The results for the coordination codes are shown in Figure 3. Notably, the ideal amount of coordination during a collaborative task should be in the mid-range of percentage of time [17]. Too little coordination could indicate a lack of collaboration, and too much coordination could indicate that the coordination is not working as intended [17].

4.1.1 Coordination of People. Figure 3 shows that Group 1 coordinated people 19% more during their RoboLoom build. Coordination of people occurred when tasks (1) were parallel (DP4), (2) were physically difficult (DP3), (3) had physicality requiring simultaneous manipulation (DP5), (4) had non-specific hardware (DP2), and (5) were repetitious (DP1).

Group 1 began by splitting into pairs to parallelize the assembly of RoboLoom's frame (Appendix A, Figure 14) and motor assemblies (Appendix A, Figure 15), necessitating coordination of pairs and task assignments. As they continued, they coordinated people for the new set of tasks, shown in Figure 2. The Ashford Loom has a more serial approach to the frame (Appendix A, Figure 17c) and yarn manipulation portions of the machine (Appendix A, Figure 17b). We hypothesize that this difference in serial assembly method led to fewer opportunities for coordination of people.

Physically difficult tasks allowed for coordination of people through specialization. For example, P1 struggled with dexterous tasks (aligning a T-Nut, Figures 9 and Appendix A, Figure 14c and attaching RoboLoom's length beam, Appendix A, Figure 14b) and P4 took over. Additionally, tasks requiring larger forces provided the opportunity to coordinate people. For example, P2 took over for P3 as she tried, but failed to lift the castle (Appendix A, Figure 17b) of the Ashford Loom. Both looms had difficult hardware, but we observed more instances in RoboLoom.

Physicality requiring simultaneous manipulation prompted coordination of people as well. For example, aligning RoboLoom's heddle guide (Appendix A, Figure 15a) requires 40 heddles to be manipulated simultaneously, leading P1 and P4 to coordinate. A similar event occurred on the Ashford Loom, when P2 asked for help holding the beater (Appendix A, Figure 17d). Fewer such moments were observed on the Ashford due to less simultaneous manipulation.

Non-specific hardware introduced uncertainty, so participants coordinated to have group members check their work. For example, P1 asked P4 to double-check a measurement she took, requiring coordination of people.

Finally, repetitive hardware created the possibility for coordinating people. RoboLoom requires 46 corner assemblies (Appendix A, Figure 14c), so P2 coordinated with her group to split the work for speed. While individually accomplishable, the potential for optimization was reason enough for P2 to coordinate more help.

4.1.2 Coordination of Materials. Figure 3 shows that Group 2 increased their coordination of materials by 20% during RoboLoom assembly. This occurred when tasks were mentally difficult (DP3) or during parallel (DP4) tasks to share tools.

RoboLoom uses many unfamiliar hardware components. To reduce mental difficulty, P6 coordinated the organization of the materials. This happened frequently in the RoboLoom assembly (15/20 data points). This occurred less in the Ashford Loom assembly (2/4) as participants were more familiar with the materials, and materials were labeled for each step.

Group 2 also coordinated materials when they had multiple, short, individual, parallelizable tasks that required the use of the same tool during RoboLoom's assembly. The Ashford Loom's task required more time, and participants did not coordinate sharing the tool; instead, they serialized their assembly.

4.1.3 Coordination of Specific Tasks. Group 1 increased coordination for specific tasks during their RoboLoom build by 24% (Figure 3). Group 1 coordinated specific tasks during their RoboLoom build because tasks (1) were non-specific (DP2), (2) were mentally difficult (DP3), (3) were repetitive (DP1), (4) could be optimized for time or ease through parallelizability (DP4) or non-specificity (DP2), (5) had physicality requiring simultaneous manipulation (DP5), or (6) had physicality involving manipulating the same object (DP5). During Group 1's Ashford Loom assembly, the participants coordinated on specific tasks where tasks (1) were non-specific (DP2), (3) were repetitive (DP1), or (5) had physicality requiring simultaneous manipulation (DP5).

Coordination of specific tasks was observed when hardware was non-specific and could be assembled incorrectly. For example, RoboLoom's frame requires that the corner assemblies be built in a specific orientation, but this is not enforced during their assembly (Appendix A, Figure 14c). This was also observed in the Ashford Loom assembly during the calibration of the shafts (Appendix A, Figure 18b). The hardware did not specify correctness of assembly, leaving participants unsure and seeking confirmation from the group. These instances of non-specific hardware created opportunities for coordination of specific tasks.

Coordination of specific tasks was also observed with unfamiliar hardware, causing mental difficulty. For example, P2 and P3 were unfamiliar with mounting RoboLoom's heddles to the motors (Appendix A, Figure 15c), so they coordinated while they figured out how to achieve the task. Once they had a solution, they stopped coordinating. The Ashford Loom's hardware was familiar already (due to hardware choices like screw selection, etc.), so no instances of coordinating specific tasks occurred for this reason.

Similarly, when tasks were repetitive, participants would split the work and initially coordinate how to complete the task, then stop coordinating once they had completed the first few instances of the repetitive task (e.g. making corner assemblies, (Appendix A, Figure 14c), or making shafts, Appendix A, Figure 18a).

Optimizable assembly provided another opportunity for coordination. Group 1 chose to parallelize RoboLoom's assembly, allowing them to coordinate specific tasks and their scheduling. This occurred multiple times throughout the build as tasks were finished, as shown in Figure 2. Task scheduling was not observed in the Ashford Loom assembly, as the task order was more constrained. Another instance of optimization was non-specific hardware that allowed for multiple approaches. For example, P1 and P4 coordinated the beam attachment order in RoboLoom's assembly (Appendix A, Figure 14).

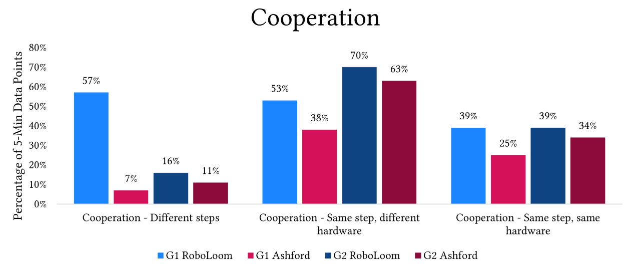


Figure 4: The percentage of 5-minute data points coded for the cooperation codes for each of the four test cases. G1 RoboLoom shows an increase in all cooperation codes over the Ashford loom, especially in different steps.

Participants also coordinated on specific tasks whose physicality necessitated the simultaneous manipulation of multiple objects. For example, when P1 and P4 were attaching the heddle guide to the RoboLoom, they coordinated the manipulation of multiple heddles into place while also holding the guide steady (Figure 11 and Appendix A, Figure 15a).

Finally, participants coordinated on specific tasks whose physicality involved manipulating the same object. For example, when P1 and P4 were assembling RoboLoom's height beams, they both needed to manipulate the base frame (Appendix A, Figure 14a). This required coordinating the loom's position and managing the forces exerted on the frame.

4.2 Cooperation

The results of the coding for cooperation codes are shown in Figure 4. We separate cooperation into three categories: different steps, same step, different hardware, and same step, same hardware. We chose these categories as a way to describe how closely participants were working together. When working on different steps, the need for cooperation may be lessened. However, during the same step, when working with different hardware, participants may cooperate to share their knowledge of the task. Finally, when physically interacting with the same hardware during the same step, more forms of cooperation may be needed.

4.2.1 Different Steps. Group 1 cooperated on different steps 50% more during their RoboLoom assembly (Figure 4) due to parallelization (DP4) of the frame (Appendix A, Figure 14) and yarn manipulation (Appendix A, Figure 15) subsystems. They split into pairs, causing the large continuous segments of the 'cooperation-different steps' code in Figure 2 without sacrificing cooperation on the same step. Group 2 discussed parallelizing their RoboLoom assembly only at the end of the build, resulting in less cooperation on different steps. Groups did not split assembly with the Ashford loom due to its serial nature.

4.2.2 Same Step, Different Hardware. Group 1 was observed spending more time (15%, Figure 4) cooperating on the same step, different hardware during RoboLoom's assembly. Participants cooperated on the same step, different hardware when tasks were repetitive

(DP1). Both looms had repeated assembly tasks due to loom symmetry. For example, when adding the length beams (Appendix A, Figure 14b) on RoboLoom, participants split the work and added them simultaneously. Other repetitive tasks (e.g. building 16 shafts for the Ashford Loom, Appendix A, Figure 18a, or making 46 corner assemblies, Appendix A, Figure 14c, or 40 motor assemblies, Appendix A, Figure 15b, for RoboLoom) were also split. RoboLoom had more repetitive tasks, which could have led to more observed instances of 'cooperation - same step, different hardware'.

4.2.3 Same Step, Same Hardware. Group 1 increased cooperation on the same task with the same hardware by 14% (Figure 4) when (1) the hardware's physicality required simultaneous manipulation (DP5) and (2) tasks were physically difficult (DP3). During both RoboLoom and Ashford Loom assemblies, tasks would require one participant to hold a part while another fastened it. RoboLoom had additional tasks where simultaneous manipulation of multiple parts was necessary, e.g. attaching the heddle guide (Appendix A, Figure 15a) or mounting the motor frames (Appendix A, Figure 15d).

Additionally, we saw more cooperation on the same hardware during RoboLoom's assembly when parts were difficult to manipulate. For example, when participants were attaching the length beam, they often needed two people to guide the beam and align the corners simultaneously (Appendix A, Figure 14b).

Some instances of this code were not attributed to aspects of the hardware, e.g. participants handing each other hardware (which happened for both looms).

4.3 Communication

The results for the communication codes are shown in Figure 5. We observed a difference in helping behaviors across loom hardware through knowledge co-creation and transfer. Though we coded for communication about troubleshooting and instruction confusion, we did not observe differences across the loom hardware in these behaviors.

4.3.1 Helping: Knowledge Co-creation. Both groups engaged in co-creating knowledge more during the RoboLoom assembly than the Ashford loom assembly. We observe that knowledge co-creation occurred when (1) hardware was non-specific (DP2), (2) hardware was mentally difficult (DP3), and (3) hardware was physically difficult (DP3). In the case of RoboLoom, there were more instances of these hardware features than the Ashford Loom, leading to more co-creation of knowledge.

Some hardware has a non-specific design, meaning it can be assembled multiple ways, only one of which is correct. Participants would engage in knowledge co-creation when hardware could be assembled wrong to discuss the function of the hardware, the correctness of their assembly, and mistakes made during the assembly. For example, participants communicated about the mounting of RoboLoom's warp beam, using their knowledge of loom function to determine the correct way to assemble the beam (Figure 16). Additionally, Group 2 built their base frame incorrectly and collaboratively created the solution to fix their error. Finally, Group 1 communicated using prior loom knowledge to co-create the knowledge of their final assembly's correctness. These opportunities for

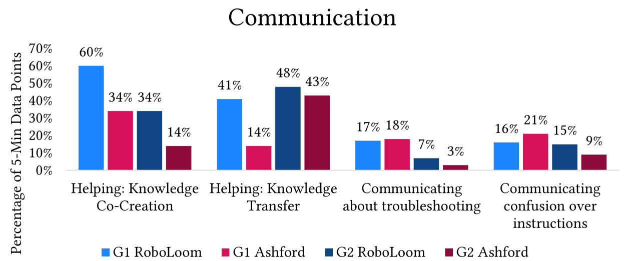


Figure 5: The percentage of 5-minute data points coded for the communication codes for each of the four test cases. The graph shows a difference in helping codes for G1's RoboLoom build.

uncertainty due to a lack of hardware specificity lead to opportunities to co-create knowledge.

Another instance of co-creating knowledge was during the mentally difficult task of using unfamiliar hardware. RoboLoom's hardware components were unfamiliar to the participants, leading to discussion on assembly and function. For example, fastening methods (e.g. corner assemblies (Appendix A, Figure 14c), heddle-to-motor attachment (Appendix A, Figure 15b), bobbin-to-holder attachment, and shaft collars) attaching length beams, and the ratchet and pawl of the warp beam were novel and led to knowledge co-creation. The Ashford loom also had unfamiliar tasks, though fewer than RoboLoom. Participants, while familiar with the use of Philips head screws, were unfamiliar with screwing into hardwood, prompting discussion on why the task differed from previous experience.

Physically difficult tasks caused participants to struggle, creating the opportunity for group problem-solving. These tasks included assembling RoboLoom's length beam (Appendix A, Figure 14b), aligning and attaching RoboLoom's motor frames (Appendix A, Figure 15c), aligning and attaching RoboLoom's heddle guide (Appendix A, Figure 15a), mounting RoboLoom's tension rods (Appendix A, Figure 16), squaring both loom frames, and screwing into the Ashford Loom's hardwood. These struggles prompted the groups to discuss reasons and solutions for the struggles, co-creating the knowledge of how to help.

4.3.2 Knowledge Transfer. Group 1 showed an increase in knowledge transfer from one participant to another during their RoboLoom assembly by 27% (Figure 5). We observed some common themes when participants transferred knowledge, including sharing understanding from the directions and sharing experience with the hardware. When participants shared knowledge from the directions, one group member would read the instructions and disseminate the information. This was not observed to be influenced by the hardware. When participants transferred knowledge about hardware, it was when hardware was (1) repetitive (DP1), (2) mentally difficult (DP3), (3) physically difficult (DP3), (4) non-specific (DP2), and (5) physically requiring simultaneous manipulation (DP5).

In the case of repetitive tasks like making corner assemblies (RoboLoom, Appendix A, Figure 14c) or shafts (Ashford Loom, Appendix A, Figure 17), participants would share tips and tricks with other participants to help with the assembly.

In the case of physically difficult hardware like the length beam assembly (RoboLoom, Appendix A, Figure 14b) or tightening screws into hardwood (Ashford loom, Appendix A, Figure 17) the participant who had figured out how to do the task would share that information to reduce group effort. This would happen in cases where one participant was able to figure it out fully on their own. In the cases where this was not possible, participants would co-create the knowledge as discussed in the previous section.

When tasks were mentally difficult due to the unfamiliarity of one participant but not another, participants would transfer knowledge to ensure more group members could complete the assembly. This was seen for RoboLoom when using a ball-end Allen key to tighten screws. This was not observed during the assembly of the Ashford Loom.

When hardware features were non-specific regarding assembly methods, participants with the knowledge of correct assembly methods would transfer this information to other participants working on the same task. For example, when P1 figured out the arrangement of 8020 in the foot holder of RoboLoom (Appendix A, Figure 14d), they shared this information with P4, who was working on this task with them. Similarly to difficulty, the instances of knowledge transfer happened when one participant created the knowledge on their own, but there were instances where knowledge was co-created (discussed previously).

Participants shared ideas with their group when hardware had to be simultaneously manipulated. For example, in both RoboLoom and Ashford Loom assemblies, one participant had their hands full with a portion of the task and knew how to complete the rest of the task, but was physically unable to do so. Participants then transferred the knowledge necessary to complete the task to another group member. When tasks (mounting motor frames – Appendix A, Figure 15d – or main frame assembly – Appendix A, Figure 17) required more than one set of hands, participants would communicate more.

4.4 Blocked Collaboration

Our results for the 'blocked collaboration' code are shown in Figure 6. Collaboration was blocked only once in both RoboLoom builds when a participant did not see the next available task. However, after discussion with the group, the participants found a task to do and continued collaborating.

Conversely, both groups had multiple instances where collaboration was attempted but not successful for the Ashford loom assembly. These instances happened for several reasons, including (1) a lack of tools, (2) hardware physicality designed for a single builder (DP5), (3) non-parallel hardware (DP4), (4) small physicality (DP5), and (5) too much mental difficulty (DP3). When participants lacked the tools to complete tasks, they would build without collaborating. This was not due to hardware design.

When the physicality of the Ashford loom was designed such that tasks did not allow for multiple builders, the participants would not collaborate on the assembly. For example, when a single bolt needed to be hammered in, P5 completed this task while P6 and P7 watched him. This task does not require multiple sets of hands or simultaneous manipulation of two or more objects and thus requires only one person to complete it.

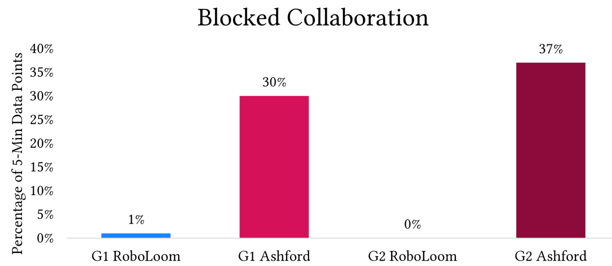


Figure 6: The percentage of 5-minute data points coded for the Blocked Collaboration code for each of the four test cases showing more blocked collaboration during the Ashford Loom building.

These instances were often coupled with the inability to move to the next task without first completing the current step of assembly (i.e., non-parallel hardware design). Without another task to complete, participants could not cooperate. For instance, we observed participants of Group 2 attempting to cooperate by going forward in the instructions while P6 attached the ratchet gears. However, upon P7 reading the next step, P5 and P7 saw they could not attach the handle to the ratchet gears until they had been attached to the loom. They then decided they could not cooperate and instead waited for P6 to finish her task.

Collaboration was also blocked by small physicality, creating a lack of space for cooperation. For instance, when inserting shafts into the castle of the Ashford Loom, P6 inserted the shafts while P7 held them in place. P5 stood by and watched as there was no room or need for his physical help.

Finally, we observed blocked collaboration when tasks were too mentally difficult, e.g., a participant did not contribute to problem-solving or troubleshooting due to a lack of building knowledge.

5 Discussion

We observed that hardware design features did provide opportunities for participants to engage in collaborative behaviors. These hardware features can be summarized into 5 categories: repetitiveness, specificity, difficulty, parallelizability, and physicality.

5.1 Repetitiveness

Repetitive hardware elements – e.g., Figure 7 – can create opportunities for coordination of people and specific tasks, cooperation on different hardware, and communication to co-create and transfer knowledge.

When tasks are repetitive, participants can coordinate to divide the labor of a repetitive task and then coordinate during the specific task. Participants can then cooperate on their shared task to decrease individual effort and overall assembly time. Throughout the assembly, participants can communicate to co-create the knowledge of how to assemble their shared repetitive task. When a new participant joins the repetitive task, or one participant discovers something about the task, there is opportunity for knowledge transfer as the participants share tips.

Longer repetitive tasks increased the number of cooperation-coded data points, as participants spent longer on the task. However,

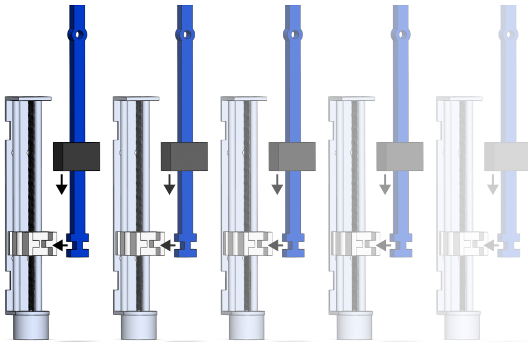


Figure 7: A visual of a task whose repetitive nature can provide the opportunity for a divide and conquer approach to assembly, allowing for coordination, cooperation, and communication.

we did not observe more coordination or knowledge transfer during these instances. Further study is needed to specifically determine the effect that the length of a repetitive task has on cooperation, coordination, and knowledge co-creation and transfer, specifically in the existence of a saturation point for coordination and knowledge transfer and co-creation.

Designing educational kits to have repetitive tasks can provide the opportunity for participants to coordinate, cooperate, and transfer and co-create knowledge. Splitting repetitive tasks allows for physical cooperation while still encouraging communication about the shared goal. When designing for collaborative assembly, the length and number of repetitive tasks should be considered, as longer, more numerous tasks can provide more opportunities for cooperation; however, they may not provide more opportunities to coordinate, transfer knowledge, or co-create knowledge.

5.2 Specificity

The specificity of hardware – e.g., Figure 8 – can influence the opportunity to coordinate people and specific tasks and communicate to co-create and transfer knowledge. We define specificity as how pre-determined the hardware is for its intended purpose, similar to the concept of poka-yoke [31], or "error proofing." Specificity can refer to the directionality of a piece (i.e., can it be assembled backward), the flexibility of a piece (i.e., how many uses it has), or where it can be mounted. We observed that the amount of specificity of the hardware components affected how participants collaborated through their knowledge co-creation, knowledge transfer, and coordination of people and tasks.

Non-specific hardware features provide opportunities for participants to discover the knowledge of how the part should be assembled (e.g., does the direction matter?). The participants can co-create this knowledge using the hardware's intended function (e.g., this part should only rotate clockwise) or through trial and error in assembly. They can also transfer this knowledge to others in their group as they build. Non-specificity of hardware can cause participants to have questions as they assemble, leading to coordination of specific tasks and potentially asking group members to check their work (and coordinating people to do so).

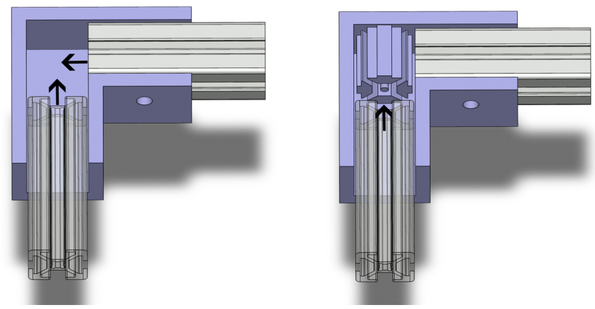


Figure 8: An example of specificity showing a design feature that can be added to a corner bracket, forcing the way aluminum extrusion mounts onto it. The specificity directs the assembly, but the non-specificity can allow opportunity for coordination and communication.

Though in commercial applications hardware is usually designed to be as error-proof and as easily assemblable as possible [33], as is the case with many hardware features of the Ashford Loom, this does not provide the opportunity for participants to collaborate. Poka-yoke hardware features are designed to be easily assembled by one person, making collaboration unnecessary. On the other hand, if hardware is too non-specific with little way to determine correctness, participants may turn to facilitators for help, thus preventing collaboration.

When designing for collaboration, non-specific hardware features can be used to give opportunities for more communication in the form of knowledge co-creation and knowledge transfer. However, this provides more opportunities for assembly errors. While this in itself can be used as a learning opportunity, it must be carefully balanced to prevent frustration from discouraging participants. When designing an assemblable educational toolkit, specificity can be used to increase opportunity for communication (knowledge co-creation or transfer) and coordination (people and tasks), but must be carefully considered with time and frustration in mind.

5.3 Difficulty

Difficult hardware – e.g., Figure 9 – can create opportunities for coordination, cooperation, and communication, but when hardware is too difficult, it can inhibit collaboration. The difficulty of the hardware refers to how difficult it is for participants to work with the hardware. This refers to mental difficulty in how familiar a participant is with the hardware or physical difficulty in how much effort or dexterity the task takes. Mentally difficult tasks affected the coordination of specific tasks and knowledge co-creation and transfer. Physically difficult tasks affected the coordination of people, the cooperation on the same hardware, and knowledge co-creation and transfer.

Physically difficult tasks like attaching the length beam of RoboLoom gave participants an opportunity to coordinate people and cooperate on the task. Participants found the length beam of RoboLoom difficult to maneuver, align with the corner fasteners, and fasten down. This allowed for the opportunity to coordinate multiple people to work on the task, which then led to cooperation

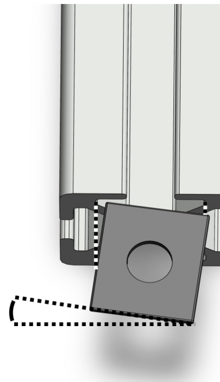


Figure 9: An example of difficulty showing the small tolerance for inserting a t-nut into aluminum extrusion, requiring dexterous manipulation to successfully assemble. Difficult hardware can create opportunity for coordination, cooperation, and communication, but can also prevent collaboration if the task is too difficult.

on the task. Conversely, when some tasks were too physically difficult for participants, such as aligning height beams, they would take the opportunity to coordinate their team and assign the most physically skilled at this portion of the assembly to the task, while finding other roles for the other team members.

When tasks were mentally difficult, we observed the coordination of specific tasks and knowledge co-creation and transfer. For example, participants did not coordinate when they were working with Philips head screws. Commercial designs for assemblable devices privilege ease of assembly to capture a wider audience of abilities. This leads to the use of familiar fastening devices. However, when using unfamiliar fastening methods, participants would coordinate about the task, as they did not already have a mental model of the necessary steps. This increase in difficulty, while non-optimal in a commercial design, provided the opportunity for collaboration. However, when tasks were too difficult and no participants could find a solution, they would stop collaborating and ask a facilitator for help.

When designing assemblable hardware, features can provide the opportunity for coordination of people and specific tasks, cooperation, and co-creation and transfer of knowledge. Hardware features that are physically and mentally difficult provide these opportunities. However, there are other effects of difficulty that should be considered when designing such hardware. For example, the possible frustration participants could face may discourage them from completing the task, or a too difficult task could result in participants needing help rather than being able to collaborate to find the answer. When designing purposeful difficulty, the level of difficulty must be balanced so that participants are not discouraged and can collaboratively solve the problem.

When hardware elements were parallelizable – e.g., Figure 10 – cooperation on different steps and cooperation on the same step were possible. There was also the opportunity for coordination of people, materials, and specific tasks.

When hardware was parallelizable, groups of four would split into pairs to optimize the assembly for time. This provided the opportunity to not only cooperate on different steps, but also to

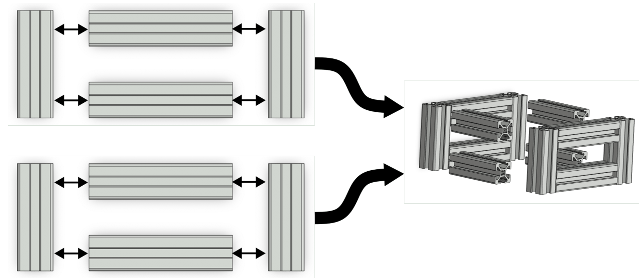


Figure 10: An example of a box with the ability to be built in parallel, one side at a time, then the sides are assembled together. This parallelizability allows the opportunity for coordination and cooperation.

cooperate on the same step within pairs (when allowable by the hardware, see physicality). This split also allowed coordinating people and specific tasks. In groups of three, parallelizable hardware led to groups splitting into one pair and one individual. The individual would then be cooperating on a different step, but not directly communicating with group members.

When parallel tasks had shared steps or materials, it created the opportunity for the pairs to come back together and coordinate the shared materials, and also to transfer knowledge they had gained during their task building.

5.4 Parallelizability

When designing educational kits, parallelization can be used to increase cooperation, but planned dependencies can create opportunity for groups to come back together to coordinate and communicate. While fully parallelizable tasks increase cooperation and speed of assembly, they could limit communication between group members if tasks do not overlap at all. In an educational setting, this could lead to inaccurate beliefs about the contribution of others and less co-creating knowledge. This may have adverse effects on group dynamics, attitudes, and collaborative educational experiences. Further study is needed into the effect of parallelizability on cooperation and communication to determine the trade-offs of speed and perceived collaboration.

5.5 Physicality

The physicality of hardware – e.g., Figure 11 – can provide the opportunity for coordination, cooperation, and communication to transfer knowledge. The physicality of a device is a feature that we use to refer to the size of the hardware, how many non-fixed parts the hardware has while it's being assembled, and how many components interact with each other during an assembly task.

The size of hardware can affect cooperation on the same step, same hardware. When the hardware is too small for multiple people to work, it can block attempts at cooperation, potentially leading to adverse effects on collaborative attitudes.

When assembly tasks require simultaneous manipulation or multiple sets of hands to complete, this creates the opportunity to coordinate the specific task and cooperate on the same step, same hardware. Multiple moving parts add complexity and are thus usually avoided in designing commercial assemblable devices.

Figure 11: An example of physicality, where the alignment of heddles with the guide requires manipulating multiple objects simultaneously to keep them aligned as the guide is put over the top. This can provide opportunities for coordination, cooperation, and communication.

However, this added complexity provides the opportunity for participants to manage what was going on to ensure smooth assembly. When more parts need to be manipulated than one participant can manage alone, participants can cooperate on these steps to fill the need for help. Designing hardware features to require multiple simultaneous manipulations can encourage assemblers to coordinate and cooperate.

Physicality, where different pieces attach to the same piece of hardware, provided the opportunity for participants to coordinate specific tasks as they worked on the same piece, communicating their various needs for orientations, movements, or forces exerted at different times. During this process, the opportunity to communicate to transfer knowledge about tasks and assembly also arises. This additional complexity is usually avoided in commercial assemblies as it adds effort to the assembly process. However, this effort increases the opportunity for coordinating tasks and transferring knowledge.

When designing assemblable kits, the physicality of the hardware should be considered, as it can provide the opportunity for coordination, cooperation, and communication to transfer knowledge. Increasing simultaneous manipulations and requiring manipulation of the same piece for different tasks adds complexity to the assembly, shifting it away from commercial design principles, which optimize for ease of assembly. These features can provide the opportunity for collaboration, but could also increase frustration, potentially to the point of detriment to collaboration. Further study is needed to understand the trade-off in designing for physicality and the impacts on collaboration and frustration.

5.6 Summary

We have discussed five design principles for nuanced interactions during collaboration, which should be carefully considered for the context of the assembly. These principles (repetitiveness, specificity, difficulty, parallelizability, physicality) have been studied in individual building contexts for error reduction, ease of building, or speed of building, but have not been studied in a collaborative context. Entirely parallel building may be faster and easier, but if students do not interact with each other, they may have negative learning outcomes [6]. Depending on the end goal of the assembly (speed, educational aspect, cooperative building, system integration learning), designers may want to change how much parallelization is used in design. In a collaborative educational experience, if the designs are too repetitive, they might be less difficult and thus decrease the communication among participants. We argue that each of these design principles is a continuous axis to consider when designing hardware. These principles can influence collaborative behaviors in the initial ways we present here, but may also have more nuanced interactions with each other that require further study.

There are potential effects of these hardware design principles on frustration (difficulty, repetitiveness, and physicality) and how this could influence collaboration and participation. We saw that these design principles, when made more challenging, lead to increased collaboration. However, with the increase of challenge comes the risk of frustration, which could adversely affect a student's collaborative experience. For example, designers should not make tasks so repetitive that they frustrate users to the point of giving up; tasks should be difficult enough that users ask each other questions and collaborate, but not so difficult that they cannot work. Future work should examine the frustration-challenge balance with these design principles.

6 Conclusion

In this paper, we presented a crossover study involving two groups of participants building two looms: RoboLoom and the Ashford Loom. These devices were specifically chosen to investigate the hardware design features that influence collaborative actions during the assembly process. Rather than focusing on task design, we utilized the 3Cs framework to observe and categorize collaborative behaviors, examining the engaged hardware to determine its potential impact on group collaboration in a collaborative engineering setting.

Our observations and analysis revealed that different hardware designs facilitated distinct collaborative actions in the loom-building process. We identified five key categories of hardware features: repetitiveness, specificity, difficulty, parallelizability, and physicality.

- DP1. **Repetition** of hardware elements provided opportunities for coordination of people and tasks, cooperation on the same step but different hardware, and communication to transfer knowledge.
- DP2. **Specificity**, or features to indicate or error-proof assembly, influenced the opportunity for communication to co-create or transfer knowledge. Hardware that was highly "error-proof" reduced the need for assistance, whereas open-ended hardware often required external help from instructors, limiting peer-to-peer assistance. Mid-range specificity fostered more collaborative actions through co-creation or transfer of knowledge.
- DP3. **Difficulty**, both mental and physical, mirrored the effects of specificity. An optimal level of difficulty encouraged communication to co-create or transfer knowledge and the coordination of people, materials, and tasks. However, tasks that were too easy or too challenging reduced such interactions.

DP4. **Parallelizability** in hardware created the opportunities for cooperative behaviors.

DP5. **Physicality**, both in size and the requirement of simultaneous manipulation, created opportunities for coordination of people and tasks, cooperation on the same hardware pieces, and communication to transfer knowledge. Smaller hardware was observed to prevent physical collaborative actions.

Collaborative work is beneficial for engineering education [18, 26, 29, 44]. One opportunity to work collaboratively in engineering education is through the assembly of hardware kits. We posit that by viewing the assembly of the kit as its own sort of tangible user interface, the above-presented design features, when used as a framework to design hardware for assembly, can facilitate a collaborative user experience. By designing hardware with these features in mind, educators can create opportunities for coordination, cooperation, and communication. Specifically:

C1. Coordination can be facilitated by hardware that is parallelizable, physically and mentally challenging, requires simultaneous manipulation, semi-specific, and repetitive.

C2. Cooperation can be enhanced by hardware that is parallelizable, is physically challenging, requires simultaneous manipulation, or is repetitive.

C3. Communication can be encouraged by hardware that is physically and mentally challenging, requires simultaneous manipulation, is semi-specific, or is repetitive.

C4. Conversely, hardware that is non-parallel, is overly mentally challenging, is small, or does not require simultaneous manipulation of more than two objects can block collaboration when it is attempted and possibly discourage future attempts to collaborate.

In this paper, we presented our initial findings on these hardware feature categories as a foundation for design recommendations and further research on facilitating collaboration through interactions with hardware kits. As our study was limited in number and diversity of participants (most coming from Informatics), future work should include further study of these design features and interactions with them from different populations. Future work should also include a causal study of these design features, designing for and against them to assess their impact on collaboration. Finally, this paper presents an initial point for exploring design features for collaboration with many open avenues for further exploration into different collaborative contexts, further study into the interplay of these features and their effects on collaboration, and studies on challenge-frustration trade-offs.

Acknowledgments

This work was supported by National Science Foundation grant 2100401.

References

[1] National Academies. 2012. *Education for Life and Work: Developing Transferable Knowledge Skills in the 21st Century.* National Academies Press, Washington, D.C. https://doi.org/10.17226/13398

[2] National Academies. 2013. *Next Generation Science Standards. For States, By States.* National Academies Press, Washington, D.C. https://doi.org/10.17226/18290

[3] National Academies. 2017. *A New Vision for Center-Based Engineering Research.* National Academies Press, Washington, D.C. https://doi.org/10.17226/24767

[4] Diana Africano, Sara Berg, Kent Lindbergh, Peter Lundholm, Fredrik Nilbrink, and Anna Persson. 2004. Designing tangible interfaces for children's collaboration. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems (CHI EA '04)*. Association for Computing Machinery, New York, NY, USA, 853–868. https://doi.org/10.1145/985921.985945

[5] Michael J. Baker. 2015. Collaboration in collaborative learning. *Interaction Studies* 16, 3 (Jan 2015), 451–473. https://doi.org/10.1075/is.16.3.05bak

[6] Brigid Barron. 2003. When Smart Groups Fail. *The Journal of the Learning Sciences* 12, 3 (2003), 307–359. http://www.jstor.org/stable/1466921

[7] Paulo Blikstein. 2015. Computationally Enhanced Toolkits for Children: Historical Review and a Framework for Future Design. *Foundations and Trends® in Human–Computer Interaction* 9, 1 (2015), 1–68. https://doi.org/10.1561/1100000057

[8] Carlos Boya-Lara, Doris Saavedra, Aaron Fehrenbach, and Angel Marquez-Araque. 2022. Development of a course based on BEAM robots to enhance STEM learning in electrical, electronic, and mechanical domains. *International Journal of Educational Technology in Higher Education* 19, 1 (Feb. 2022), 7. https://doi.org/10.1186/s41239-021-00311-9

[9] Michelene T. H. Chi and Ruth Wylie. 2014. The ICAP Framework: Linking Cognitive Engagement to Active Learning Outcomes. *Educational Psychologist* 49, 4 (Oct. 2014), 219–243. https://doi.org/10.1080/00461520.2014.965823

[10] Pierre Dillenbourg, Michael J. Baker, Agnès Blaye, and Claire O'Malley. 1995. The evolution of research on collaborative learning. In *Learning in Humans and Machine: Towards an interdisciplinary learning science.*, Spada, E., Reiman, and P. (Eds.). Elsevier, Oxford, 189–211. https://telearn.hal.science/hal-00190626

[11] Pierre Dillenbourg, Michael J. Baker, Agnès Blaye, and Claire O'Malley. 1995. *The evolution of research on collaborative learning.* Elsevier, Oxford, 189. https://telearn.hal.science/hal-00190626

[12] Pierre Dillenbourg and Michael Evans. 2011. Interactive tabletops in education. *International Journal of Computer-Supported Collaborative Learning* 6, 4 (Dec. 2011), 491–514. https://doi.org/10.1007/s11412-011-9127-7

[13] Mark Elliott. 2007. *Digital Library Of The Commons.* Ph. D. Dissertation. Indiana University. https://dlc.dlib.indiana.edu/dlc/handle/10535/3574

[14] Clarence A. Ellis, Simon J. Gibbs, and Gail Rein. 1991. Groupware: some issues and experiences. *Commun. ACM* 34, 1 (Jan 1991), 39–58. https://doi.org/10.1145/99977.99987

[15] Regina Nur Fitriyaningsih, Cucuk Wawan Budiyanto, and Rosihan Ari Yuana. 2019. Behavioral patterns of vocational students in Lego Mindstorm: A literature review. *AIP Conference Proceedings* 2114, 1 (June 2019), 060003. https://doi.org/10.1063/1.5112474

[16] Harry W. Fox and M. Renee Overton. 2006. USING ROBOTICS TO ENHANCE OUTREACH PROGRAMS FOR HIGH SCHOOL STUDENTS. *American Society for Engineering Education Illinois-Indianna and Northe Centeral Joint Section Conference* (April 2006). http://ilin.asee.org/Conference2006program/Papers/Fox-P67.pdf

[17] Hugo Fuks, Alberto Raposo, Marco Aurelio Gerosa, Mariano Pimentel, Denise Filippo, and Carlos Lucena. 2008. Inter- and intra-relationships between communication coordination and cooperation in the scope of the 3C Collaboration Model. In *2008 12th International Conference on Computer Supported Cooperative Work in Design*. 148–153. https://doi.org/10.1109/CSCWD.2008.4536971

[18] Ozdemir Gol and Andrew Nafalski. 2007. Collaborative learning in engineering education. *Global Journal of Engineering Education* 11 (2007). Issue 2.

[19] Otmar Hilliges, Lucia Terrenghi, Sebastian Boring, David Kim, Hendrik Richter, and Andreas Butz. 2007. Designing for collaborative creative problem solving. In *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition (C&C '07)*. Association for Computing Machinery, New York, NY, USA, 137–146. https://doi.org/10.1145/1254960.1254980

[20] Keiichi Ihara, Mehrad Faridan, Ayumi Ichikawa, Ikkaku Kawaguchi, and Ryo Suzuki. 2023. HoloBots: Augmenting Holographic Telepresence with Mobile Robots for Tangible Remote Collaboration in Mixed Reality. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 119, 12 pages. https://doi.org/10.1145/3586183.3606727

[21] Paul Kirschner, Jan-Willem Strijbos, Karel Kreijns, and Pieter Jelle Beers. 2004. Designing electronic collaborative learning environments. *Educational Technology Research and Development* 52, 3 (Sep 2004), 47–66. https://doi.org/10.1007/BF02504675

[22] P. A. Kirschner, R. L. Martens, and J. W. Strijbos. 2004. *CSCL in Higher Education?* Springer Netherlands, Dordrecht, 3–30. https://doi.org/10.1007/1-4020-7921-4_1

[23] Theodore Knoll, Amna Liaqat, and Andrés Monroy-Hernández. 2023. ARctic Escape: Promoting Social Connection, Teamwork, and Collaboration Using a Co-Located Augmented Reality Escape Room. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3544549.3585841

[24] Marjan Laal and Seyed Mohammad Ghodsi. 2012. Benefits of collaborative learning. *Procedia - Social and Behavioral Sciences* 31 (Jan. 2012), 486–490. https://doi.org/10.1016/j.sbspro.2011.12.091

[25] Thomas Marrinan, Jason Leigh, Luc Renambot, Angus Forbes, Steve Jones, and Andrew E. Johnson. 2017. Mixed Presence Collaboration using Scalable Visualizations in Heterogeneous Display Spaces. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17)*. Association for Computing Machinery, New York, NY, USA, 2236–2245. https://doi.org/10.1145/2998181.2998346

[26] Emma Mercier, Molly H Goldstein, Preethi Baligar, and Robin Jephthah Rajarathinam. 2023. Collaborative learning in engineering education. In *International handbook of engineering education research*. Routledge, 402–432.

[27] Tassos A. Mikropoulos and Ioanna Bellou. 2013. Educational Robotics as Mindtools. *Themes in Science and Technology Education* 6, 1 (2013), 5–14. ERIC Number: EJ1130925.

[28] Heloisa Moura, Dale Fahnstrom, Greg Prygrocki, and T. J. McLeish. 2009. ThinkeringSpace: Designing for Collaboration. *Visible Language* 43, 1 (2009), 46–61.

[29] Geraldine Gonzaga Nerona. 2019. Effect of collaborative learning strategies on student achivemement in various engineering courses. *International journal of engineering education* 1, 2 (2019), 114–121.

[30] Piia Näykki, Sanna Järvelä, Paul A. Kirschner, and Hanna Järvenoja. 2014. Socio-emotional conflict in collaborative learning—A process-oriented case study in a higher education context. *International Journal of Educational Research* 68 (2014), 1–14. https://doi.org/10.1016/j.ijer.2014.07.001

[31] Taiichi Ohno. [n. d.]. *Toyota Production System: Beyond Large-Scale Production*. Productivity Press. https://doi.org/10.4324/9780429273018

[32] Reinhard Pekrun. [n. d.]. Emotions and Learning. Educational Practices Series-24. ERIC Number: ED560531.

[33] Kylie Peppler and Yasmin Kafai. 2014. *Transparency Reconsidered: Creative, Critical, and Connected Making*. 179–188.

[34] Mario Riojas, Susan Lysecky, and Jerzy Rozenblit. 2012. Educational Technologies for Precollege Engineering Education. *IEEE Transactions on Learning Technologies* 5, 1 (2012), 20–37. https://doi.org/10.1109/TLT.2011.16

[35] Filipa Rocha, Filipa Correia, Isabel Neto, Ana Cristina Pires, João Guerreiro, Tiago Guerreiro, and Hugo Nicolau. 2023. Coding Together: On Co-located and Remote Collaboration between Children with Mixed-Visual Abilities. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3544548.3581261

[36] Jeremy Roschelle and Stephanie D. Teasley. 1995. The Construction of Shared Knowledge in Collaborative Problem Solving. In *Computer Supported Collaborative Learning (NATO ASI Series)*, Claire O'Malley (Ed.). Springer, Berlin, Heidelberg, 69–97. https://doi.org/10.1007/978-3-642-85098-1_5

[37] J. Ruiz-del Solar and R. Aviles. 2004. Robotics courses for children as a motivation tool: the Chilean experience. *IEEE Transactions on Education* 47, 4 (2004), 474–480. https://doi.org/10.1109/TE.2004.825063

[38] Bertrand Schneider and Paulo Blikstein. 2018. Tangible User Interfaces and Contrasting Cases as a Preparation for Future Learning. *Journal of Science Education and Technology* 27, 4 (Aug. 2018), 369–384. https://doi.org/10.1007/s10956-018-9730-8

[39] Ludwig Sidenmark, Tianyu Zhang, Leen Al Lababidi, Jiannan Li, and Tovi Grossman. 2024. Desk2Desk: Optimization-based Mixed Reality Workspace Integration for Remote Side-by-side Collaboration. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 44, 15 pages. https://doi.org/10.1145/3654777.3676339

[40] Samantha Speer, Ana P Garcia-Alonzo, Joey Huang, Nickolina Yankova, Carolyn Rosé, Kylie A Peppler, James Mccann, and Melisa Orta Martinez. 2023. SPEERLoom: An Open-Source Loom Kit for Interdisciplinary Engagement in Math, Engineering, and Textiles. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 93, 15 pages. https://doi.org/10.1145/3586183.3606724

[41] Gerry Stahl. 2013. *Theories of Collaborative Cognition: Foundations for CSCL and CSCW Together*. Springer US, Boston, MA, 43–63. https://doi.org/10.1007/978-1-4614-1740-8_3

[42] Danaë Stanton, Victor Bayon, Camilla Abnett, Sue Cobb, and Clair O'Malley. 2002. The effect of tangible interfaces on children's collaborative behaviour. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems* (Minneapolis, Minnesota, USA) *(CHI EA '02)*. Association for Computing Machinery, New York, NY, USA, 820–821. https://doi.org/10.1145/506443.506614

[43] Gary A. Stewardson, Trevor P. Robinson, Joseph S. Furse, and Michael L. Pate. 2019. Investigating the relationship between VEX robotics and student self-efficacy: an initial look. *International Journal of Technology and Design Education* 29, 4 (Sept. 2019), 877–896. https://doi.org/10.1007/s10798-018-9461-4

[44] Glenda S. Stump, Jonathan C. Hilpert, Jenefer Husman, Wen-ting Chung, and Wonsik Kim. 2011. Collaborative Learning in Engineering Students: Gender and Achievement. *Journal of Engineering Education* 100, 3 (2011), 475–497. https://doi.org/10.1002/j.2168-9830.2011.tb00023.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/j.2168-9830.2011.tb00023.x

[45] Daniel D. Suthers. 2012. *Computer-Supported Collaborative Learning*. Springer US, Boston, MA, 719–722. https://doi.org/10.1007/978-1-4419-1428-6_389

[46] Mihaela Taranu, Gökçe Elif Baykal, Olof Torgersson, and Eva Eriksson. 2025. Technology-Mediated Collaboration among Children with Special Educational Needs: Definitions and Measurements. In *Proceedings of the 24th Interaction Design and Children (IDC '25)*. Association for Computing Machinery, New York, NY, USA, 375–394. https://doi.org/10.1145/3713043.3728846

[47] Wahyuddin, Ernawati, Sri Satriani, and Nursakiah. 2022. The Application of Collaborative Learning Model to Improve Student's 4Cs Skills. *Anatolian Journal of Education* 7, 1 (April 2022), 93–102. ERIC Number: EJ1333566.

[48] Ashford Wheels and Looms. 2024. *Sixteen Shaft Table Loom*. https://www.ashford.co.nz/product/sixteen-shaft-table-loom/

A Loom Assemblies

A.1 RoboLoom

The assembly instructions for RoboLoom were presented in two forms: a written document and a series of videos. The written document can be found on the RoboLoom website at https://sites.google.com/view/speerloom/assembly-guide. A screenshot from one of the assembly videos with the example instruction text is shown in Figure 12.

RoboLoom (Figure 13) is designed with three major subsystems: the main frame of the loom, the motors, and the tensioning system. Each of these subsystems was designed to be built separately for most of the assembly, then attached to the main frame as part of the final assembly of motors and the tensioning system. This design was chosen to allow for simultaneous work and decrease building time for students [40].

The main frame of RoboLoom is made from slotted aluminum extrusion and primarily held together through off-the-shelf corner brackets, t-nuts, and machine screws, except for a few custom 3D-printed brackets (Figure 14a). The base of the main frame is assembled with custom 3D-printed foot holders that attach to the aluminum extrusion through machine screws and t-nuts. The foot holders are designed with an open path such that the aluminum extrusion can be placed in one of two orientations, as shown in Figure 14d. However, the loom is designed such that only one of the orientations is correct. Once the base is built, the height beams are attached to RoboLoom using corner assemblies. Corner assemblies are corner brackets pre-assembled with screws and t-nuts as shown in Figure 14c. These assemblies need to be carefully done such that the t-nut is facing the appropriate direction to slide into the slotted aluminum extrusion. T-nuts must also be carefully tightened such that they are fixed onto the screw, but not too tightly to slide into the slotted aluminum extrusion. Different lengths of aluminum extrusion are then used as the height beams to mount various components of the loom. To assemble, height beams are aligned with the corner assemblies and fixed down symmetrically on either side of the loom. Finally to ensure the frame is sturdy enough to withhold tension, a length beam is attached on each side of the loom to each of the four height beams of that side of the loom by aligning each of the corner assemblies on the height beams as the length beam is slid across the loom (shown in Figure 14b).

The motor subsystem of RoboLoom houses the motors and heddles to actuate the yarns as shown in Figure 15. There are 40 motors for each of the yarns in the loom. Each motor is part of a motor assembly that consists of the motor, the heddle (which holds the yarn), and a 3D-printed attachment mechanism shown in Figure 15b. Once assembled, each motor assembly gets mounted onto a frame that holds 10 motors (shown in Figure 15c). Each frame is then mounted to the loom simultaneously by sandwiching the frames between two slotted mounting plates attached directly to the main frame (Figure 15d). Once in place, a heddle guide (laser cut from acrylic) is added to the loom, which requires aligning each of the 40 heddles with a corresponding hole in the heddle guide plate.

The tensioning system and warp beam of RoboLoom (Figure 16) hold the warp yarns in the loom and ensure they are tensioned properly. These subsystems consist of the tensioning cage and the warp beam. The tensioning cage holds tensioning rods that have tensioning spacers with springs and disks in them. The rods must be mounted carefully by first aligning the rod through one side of the mount, moving the rod to the edge of the tensioning spacer, then aligning the rod with the other mount and sliding it through the second mount. The warp beam consists of a beam that is held in place with a ratchet gear and pawl, allowing the warp yarns tied down onto the beam to be rolled up in one direction, but not unrolled, so tension is held.

Figure 12: A screenshot from the RoboLoom assembly instructions showing a step of assembly. The written text document for assembly lists this instruction as "Put the holders onto the bottom square pieces... Put a screw through the holes and each side of the foot holder and screw on a t-nut to each screw. Slide the 8020 into the holder."

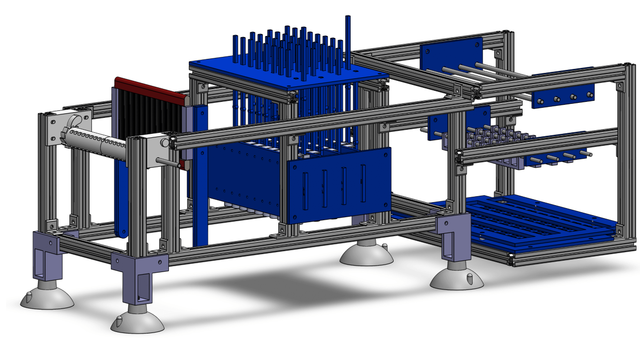
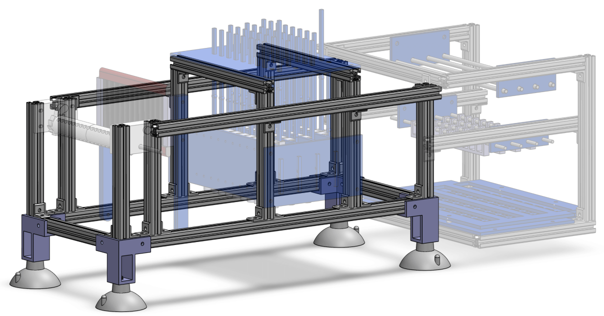
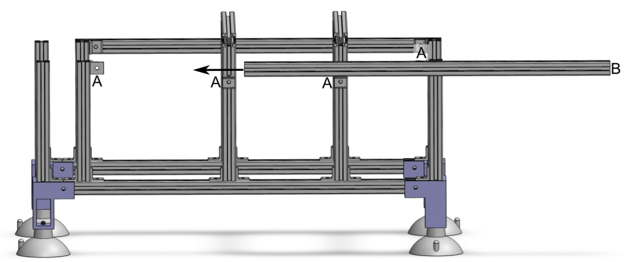


Figure 13: RoboLoom. Designed with subsystems to be assembled in collaborative groups [40].

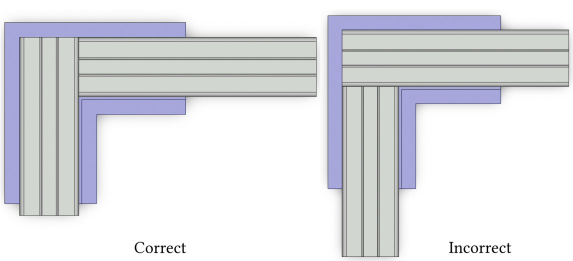


(a) RoboLoom with the main frame highlighted. The main frame consists of the aluminum extrusion used to provide the load-bearing structure to the loom, as well as off-the-shelf and custom brackets.



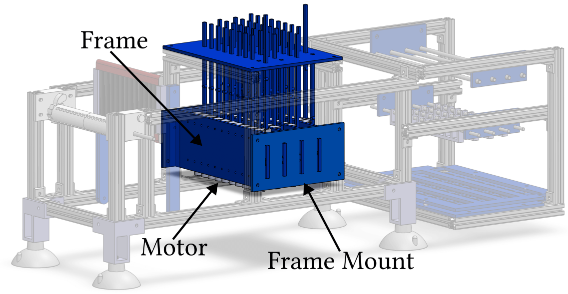
(b) The attachment of the length beam. At each of the attachment points (A), the corner assembly must be aligned as the length beam (B) is slid across the loom during assembly.

(c) A corner assembly which consists of an off-the-shelf corner bracket, machine screws, and t-nuts pre-assembled to make attachment easier.

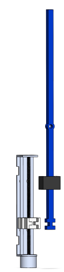


(d) The possible configurations of aluminum extrusion in the foot holders.

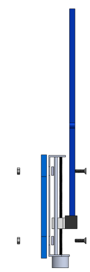
Figure 14: Components of RoboLoom's main frame.



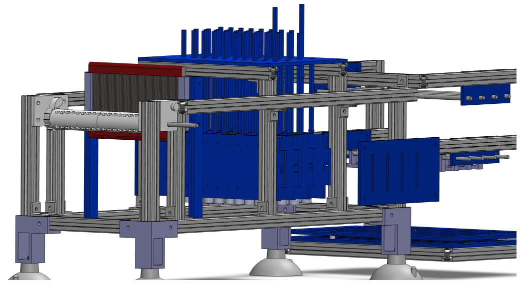
(a) RoboLoom with the motor subsystem highlighted. It consists of 10 motor assemblies mounted to each frame and the mounts for the motor frames.



(b) A motor assembly



(c) The motor attachment to the frame.



(d) The frame mount and how it attaches to the loom

Figure 15: Components of RoboLoom's motor subassembly

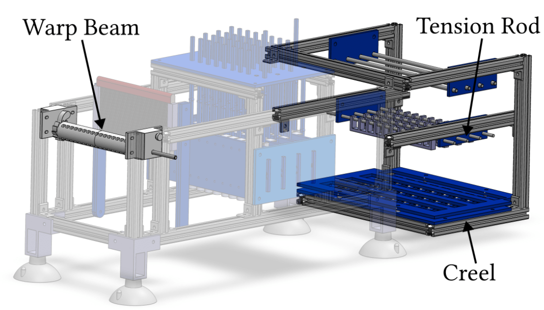
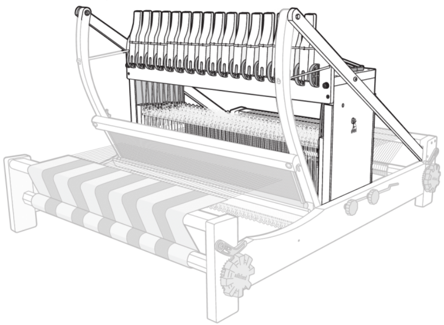
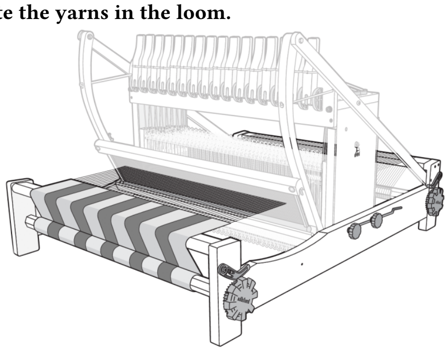


Figure 16: RoboLoom with the warp beam and tension system highlighted.

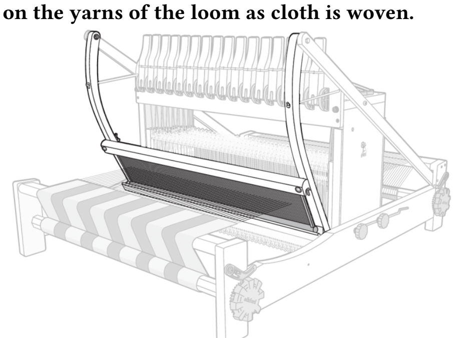
(a) The Ashford Loom [48].



(b) The castle of the Ashford Loom. It holds the shafts and levers to actuate the yarns in the loom.

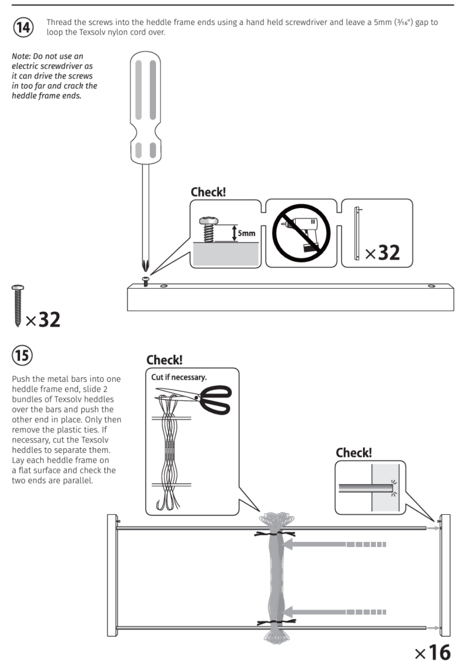


(c) The frame of the Ashford Loom. Made to hold the castle and tension on the yarns of the loom as cloth is woven.

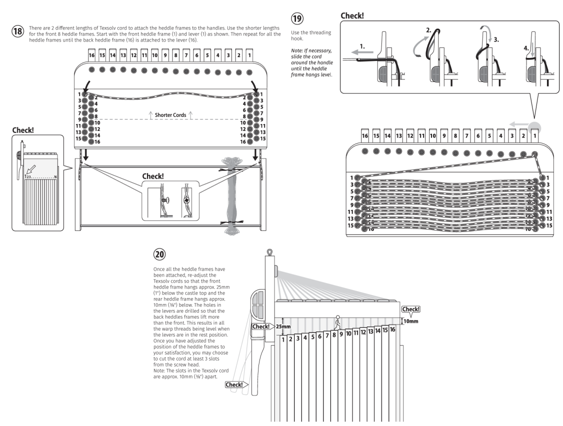


(d) The beater of the Ashford Loom, used to make cloth and suspended from the castle and frame.

Figure 17: Images of the Ashford Loom. Images taken from the publicly available Ashford loom website at https://www.ashford.co.nz/instructions/SS610.pdf [48].



(a) Instructions for the assembly of shafts in the Ashford Loom. Each shaft is made as a box with two metal sides and two wooden sides.



(b) Instructions for the threading and calibration of the shafts into the Ashford loom. Each shaft is attached by a cord, then threaded through a handle, and then calibrated.

Figure 18: Example instructions for the Ashford Loom assembly. Images from the Ashford web page, https://www.ashford.co.nz/instructions/SS610.pdf [48].

A.2 Ashford Loom

The Ashford Loom (Figure 17) is designed for hobbyist weavers to build and weave on individually. The assembly instructions for the Ashford Loom were provided online for free by Ashford. They take the form of pictorial instructions with a few words explaining each step. An example step is shown in Figure 18, and the manual

Table 3: Examples for each code of video segment descriptions that were coded. The group number, loom, and time stamp are included for each example. Looms are marked as RL (RoboLoom) or AL (Ashford Loom).

Code	Group	Loom	Timestamp (hh:mm)	Example Video Description
Coordination of People	1	RL	00:00-00:05	P1 suggests splitting into teams of two stating the sections are individual. Group agrees. P1 suggests she start with the frame because she hasn't done it. P2 suggests she take heddles. P1 suggests P2 can work alone or with someone else because there are four sets of motor frames.
Coordination of Materials	1	RL	01:50-01:55	The group coordinates passing the ruler around.
Coordination of Specific Tasks	1	AL	02:00-02:05	P3 and P2 talk about the instructions and needing to make sure certain elements are certain ways. Some conversation on how to achieve this.
Cooperation - Different Steps	2	RL	01:10-01:15	P5 works on the loom 8020, screwing things in. P6 preps the foot holders with screws.
Cooperation - Same Step, Different Hardware	2	RL	01:50-01:55	P6 and P5 each putting in height beams on different ends of the RoboLoom.
Cooperation - Same Step, Same Hardware	2	AL	00:10-00:15	P5 holds, P7 and P6 screw in.
Helping: Knowledge Co-Creation	2	AL	02:30-02:35	P6, P5, and P8 talk about where the length beam on the RoboLoom goes. P6 suggests they watch the video. P8 and P5 do and talk it through. P5 and P6 talk about doing the cross beams first.
Helping: Knowledge Transfer	1	RL	03:40-03:45	P4 watches the instructional videos with P1. P1 helps P4 understand the next steps.
Communicating About Troubleshooting	2	RL	03:10-03:15	P8 looks at the bottom and notices one beam is shorter than the other. The group talks through this mistake and tries to figure out what they need to do to fix it.
Communicating Instructions Confusion	2	AL	01:25-01:30	P7 asks P6 about the instructions. P6 explains the instructions to P7. P6 uses the materials to indicate her meaning.
Blocked Collaboration	1	RL	01:35-01:40	P4 asks if P3 feels redundant. P3 says yes a bit. P3 doesn't have a hands-on task right now.

can be found on Ashford's website at https://www.ashford.co.nz/instructions/SS610.pdf.

The loom is built from hardwood to ensure durability and sturdiness, and common hardware, to be accessible for building by hobbyists. It is structured with a main frame (Figure 17c) that holds the other components of the loom as well as the warp yarns, holding the yarn under tension to allow for proper weaving.

Inside the main frame is the castle of the Ashford Loom (Figure 17b), which is designed to hold the 16 shafts of the loom, which manipulate the yarns in the loom for weaving. The castle also has levers with ropes tied from them to the shafts to allow for manual actuation of each shaft (and consequently the yarns attached to that shaft). The shafts are built by inserting two metal bars into a piece of wood, threading on the heddles, then finishing the shaft with a final piece of wood, as shown in Figure 18a. They are then placed in the castle and attached by threading a cord around the respective lever and securing the cord to the shaft. The shafts are then calibrated to ensure they are at the correct height for weaving. The instructions for this process are shown in Figure 18b.

Finally, the Ashford Loom has a beater (Figure 17d) to help weavers push back their cloth as they weave, which is suspended from the frame and the castle.

B Coding Scheme

Our coding scheme includes four code categories with specific codes in each category. Examples of data points marked for each code are listed in Table 3.

C Code Percentages

Percentages for the general code were calculated by counting how many rows had one of the sub-codes for that code category, and then dividing this number by the total number of data points in that log. Percentages are listed in Table 4.

Table 4: Percentage of time each group was observed participating in each coded behavior.

	Group 1 RoboLoom	Group 2 RoboLoom	Group 1 Ashford	Group 2 Ashford
Coordination - General	69%	59%	57%	51%
Coordination of People	24%	16%	5%	17%
Coordination of Materials	31%	30%	21%	11%
Coordination of Specific Tasks	51%	34%	27%	29%
Cooperation - General	94%	89%	63%	89%
Cooperation - Different Steps	57%	16%	7%	11%
Cooperation - Same Step, Different Hardware	53%	70%	38%	63%
Cooperation - Same Step, Same Hardware	39%	39%	25%	34%
Communicating - General	86%	70%	57%	51%
Helping: Knowledge Co-Creation	60%	34%	34%	14%
Helping: Knowledge Transfer	41%	48%	14%	43%
Communicating About Troubleshooting	17%	7%	18%	3%
Communicating Instruction Confusion	16%	15%	21%	9%
Blocked Collaboration	1%	0%	30%	37%