International Journal of Engineering Education Vol. 37, No. 6, pp. 1655–1673, 2021
Printed in Great Britain
0949-149X/91 $3.00+0.00

Teaching K-6 Elementary Engineering with Robotics*

JOHN HEFFERNAN**
Tufts University, 99 North Hill Dr, Conway, MA 01341, USA. E-mail: johnsheffernan99@gmail.com, https://orcid.org/0000-0002-2677-4907

FLORENCE SULLIVAN
College of Education, W244 Furcolo Hall, 813 N. Pleasant St, University of Massachusetts, Amherst, Amherst, MA, USA 01003.
E-mail: fsullivan@educ.umass.edu

The robotics based Elementary Engineering Curriculum – used by students in this study – and other similar projects have the potential to increase the STEM pipeline but elementary engineering is not well-understood. Research is needed to understand how to teach engineering to students as their cognitive, motor, and social skills rapidly develop in elementary school. The authors conducted a cross-sectional case study of six grade 2 and six grade 6 elementary robotics students in the context of established K-6 elementary robotics curriculum. Students were videotaped doing an open-ended engineering task based on LEGO robotics using talk-aloud and clinical interview techniques. The engineering design processes were analyzed and compared. Significant differences were found in final projects and engineering design process related to the complexity of the ride they tried to build and the key skills and structural knowledge they brought to the task. Seven key factors identified consisted of three cognitive skills of cognitive flexibility, causal reasoning, and planning ability, three domain specific process skills of application of mathematics and science, engineering design process skills, and design principles of stability, scale, and the structural knowledge they had of LEGO robotics, most pointedly, LEGO connection knowledge. Implications of these findings for teachers are given.

Keywords: P12 engineering education; engineering design process; LEGO robotics; complexity; structural knowledge; design principles

1. Introduction

Educational robotics is an activity that, by nature, integrates science, mathematics, technology, and creativity. These specific affordances make robotics an especially attractive educational technology for engaging students in the engineering design process [1–3]. In the Next Generation Science Standards [4], the importance of the engineering design process for student learning is emphasized by its inclusion as a core discipline in science, with the design of solutions as a core practice of science, and the interdependence of science, technology, and engineering listed as a core cross cutting concept.

While the significance of engineering design for learning in K-12 education is emphasized in national standards, research related to young elementary students learning in engineering is sparse. While Bers has produced robust research focused on pre-school children's understanding of programming using robotics [5–7], we have less research focused on student learning through the actual physical design of robotics devices. The design phase of a robotic device is a constituent aspect of robotics activity [8]. It is in the design phase of activity that students contend, through intuition and direct feedback, with scientific (e.g., physics), mathematical (e.g., geometry and estimation) and technological (e.g., placement of sensors for optimal functioning) ideas [8, 9]. Therefore, designing robotics devices is a strong learning activity for children [1, 10].

The study reported here focuses on student learning of engineering in the elementary grades through design of a robotic device. The study took place in a school that features a robotics based elementary engineering curriculum for students for grades K-6. This curriculum [11] was developed from a constructionist perspective [12] and allows students the opportunity to engage in engineering design with robotics materials. The goal of this research is to understand how elementary aged students engage in the engineering design process in order to improve curricular offerings and pedagogical practices to further support student learning. The paper is organized in the following fashion: first we define the engineering design process and review empirical studies related to student learning while engaged in engineering design processes – these studies, necessarily, draw on research with older students as there is a dearth of studies at the elementary level. Next, we discuss the role of particular cognitive aspects of engaging in the engineering design process as theorized from research findings including: (1) cognitive flexibility; (2) planning; and (3) causal reasoning. Then we present our research question and the methods we used to address our research question. Finally, we present our research results and discussion, limita-

** Corresponding Author.

* Accepted 9 July 2021.

tions of the study, and suggestions for future research.

2. Background

2.1 Engineering Design Process Models

The engineering design process is an example of a general problem-solving process in the specific context of engineering. Engineering is defined as "the work of designing and creating large structures (such as roads and bridges) or new products or systems by using scientific methods." [13]. The application of mathematics and science to create something new to meet a human need is also a common definition of engineering [2]. Furthermore, engineering problems are defined by the inclusion of constraints. For example, safety and manufacturing cost limits are common engineering constraints [14]. In summary, we see that engineers integrate many fields in a creative but rigorous way.

One way to understand children's engineering skills is to characterize their engagement with the various stages defined by engineering design process models. There are a variety of design process models that can be used study of elementary robotics students. One typical engineering design process (EDP) model is shown in Fig. 1 [15].

For the purposes of this study, an engineering design process model based on observable behaviors, both visual and verbal, is the most useful for considering the totality of the process for robotics students. Our model is strongly based on existing models [6, 14, 15] but includes details specific to robotics tasks such as breaking out the prototype construction phase into programming and building. Since our model is foundational to our data analysis, we present it in the methods section further below.

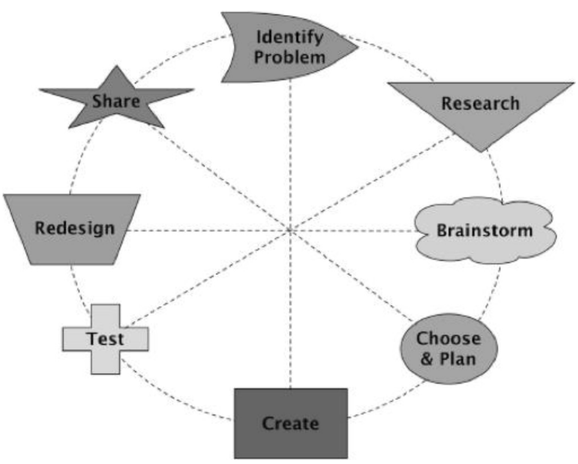

Fig. 1. Typical engineering design process model. From Dr. Meredith Portsmore, Tufts Center for Engineering Education and Outreach. Used with permission.

2.2 K-12 Engineering Design Process (EDP) Research

As noted earlier, little research on the engineering design process has been done with younger students. One exception are early studies by Roden [16, 17] who showed that very young children's collaborative problem solving strategies change over time. Other K-12 EDP researchers have found that actual design processes differ in practice from theorized, idealized, linear models [18–21]. Welch [19] found that grade seven students evaluated their design much more frequently than the theorized EDP model would predict, tried one idea at a time instead of evaluating alternatives, and preferred three-dimensional materials to two-dimensional sketches. Johnsey [20] found that grade three students jumped into making their designs prematurely.

A study by Crismond [18] revealed – in a comparison between high school students and adult engineers – that only adult expert designers use general design principles and made connections to science concepts to help their design process. The general design principles were "rules of thumb for good design" [18, p. 796] that connected the abstract to the concrete. In the realm of elementary engineering, these could be the design principles of scale, symmetry, and stability. Symmetry, in particular, has been noted as an important design aspect of building for children [22]. Crismond concluded that teachers must scaffold design tasks to help students make the connections between the concrete and the abstract. This study hopes to help teachers understand the cognitive aspects of engineering learning that lead to the ability to make design connections.

Ill- structured problems that feature manipulable materials, such as those embedded in open-ended robotics design tasks, have the potential to help students develop the cognitive abilities needed to engage in engineering design. These abilities include cognitive flexibility [23, 24], planning [25], and causal reasoning [26]. Levy & Mioduser [27] showed that complex and advanced cognition could occur in young children's interpretation of robot rules and behaviors, specifically causal reasoning. Here, we seek to investigate the nature of elementary engineering predicted by prior research. Specifically, we will investigate how cognitive flexibility, planning, and causal reasoning emerge as an aspect of elementary design process activity. Let us further define each of these constructs.

2.3 Cognitive Flexibility

Cognitive flexibility has been defined as "the ability to consider multiple bits of information or ideas at one time and actively switch between them when engaging in a task" [28, p. 26] and, more generally, as flexible thinking [29]. Cognitive flexibility can be supported through engagement in ill-structured problems [24] or when asked to invent new things [29, 30]. However, cognitive flexibility has been shown to be developmental, in that younger children lack the cognitive flexibility of older children [24, 28, 31]. For example, in our own prior work, we observed a phenomenon of "non-optimal persistence" in engineering design tasks for younger children as opposed to older children. Non-optimal persistence (or idea fixation) consists of a reluctance to start over even when it was clear that the original design idea was not working [14]. We observed this lack of cognitive flexibility in younger study participants (ages 6–7), but not older (ages 11–12).

2.4 Domain Structural Knowledge

Cutting [24] found older students were able to integrate domain knowledge into their designs (for example, recognizing the pliability of a pipe cleaner), while younger children were not. A lack of domain structural knowledge may be the cause of non-optimal persistence in children. Jonassen [26, p. 69] explains structural knowledge this way:

> ". . . domain knowledge must be well integrated in order to support problem solving. The integratedness of domain knowledge is best described as structural knowledge."

Indeed, Cutting [23, p. 115] concluded, "that without this structural knowledge, young children lacked the flexibility needed to retrieve their knowledge from memory and then coordinate it in order to solve these tool innovation tasks". Non-optimal persistence then, is related to a lack of structural knowledge, which, in turn impacts cognitive flexibility.

2.5 Planning

Previous research on the ability of elementary students to plan while engaged in engineering tasks is mixed. Some positive results were found in tightly constrained problems with familiar materials [32]. However, other studies find that young students largely skip the planning phase due to developmental constraints [33, 34]. It is possible that children can accomplish tasks ahead of projected developmental milestones in constrained tasks with familiar materials. This may not be the case for open-ended engineering challenges where knowledge transfer must occur. Planning strategies may also depend on a variety of factors such as the problem itself, student age, gender, and whether or not the student has an initial solution to the problem [32–34].

2.6 Causal Reasoning

Jonassen & Ionas [35] provide a complex model of causal reasoning and suggest different ways to support the learning of causal reasoning. In their model, problem solving and conceptual change support predictions, implications, inferences, and explanations, which, in turn, enable causal reasoning. Predictions are defined as anticipating an outcome based on the initial state of a system and plausible causal relationships. Prediction in the model is either the scientific method, namely hypothesis, or forecasting events such as weather or economic performance. Implication is defined as the same process as prediction but with more probabilistic causal relationships. Inference is further defined as the opposite process as prediction, that is, positing events and initial conditions based on a final set of conditions and plausible causal relationships. Explanation is defined as the ability to describe a system's components, functions, and causal relationships.

Engineering education provides problem-solving affordances for learning causal reasoning. Although we were unable to locate any research on causal reasoning specifically in the context of engineering design, all four enablers of causal reasoning in this model are part of engineering - predictions, inferences, explanations, and implications – but prediction and inference are the most relevant. Engineers predict how a design, process, or software program will actually function in the physical world. Inference is used when troubleshooting a model to understand why a prototype did not work, so the design can be improved.

Casual reasoning and causal inference research typically center on *a posteriori evaluation* of data to determine causes. However, engineers make *a priori* predictions of the performance of their designed systems. The predictions may be supported with simulations, models, and prototypes. In the context of LEGO robotics, students are expected to design and then build a prototype with a prediction of performance in mind and then evaluate the actual performance with respect to predicted performance.

2.7 Research Question

The aim of the study was to gain an understanding of students' skills and processes as they undertook an open-ended engineering challenge at two different ages (grade two and grade six) in the context of a K-6 LEGO robotics program [11] with the long-term goal of informing the instruction of engineering for elementary aged children. Our research question is: what do the EDPs of elementary students look like and how and why do they differ? Specifically,

- Which phases of the engineering design process do students engage in, when do they do so, and how frequently do they engage in each phase?
- If the EDPs differ, what explains the differences?
- Why are some students more successful in realizing their design ideas?

We clearly define our EDP process model in the methods section (below). In addition, we pay special attention to the cognitive factors of cognitive flexibility, planning, and causal reasoning as indicators of student learning activity and process in the open-ended design activity.

3. Methods

3.1 Research Design

The research design consists of a comparative, multiple case study. While the type of data collected (detailed below) was primarily qualitative in nature, we utilized a combination of qualitative (thematic coding) and quantitative methods (primarily frequency counts and durational measurements) to help analyze these data. Quantifying qualitative data has a long history in the learning sciences, starting with Newell & Simon [36] who developed a protocol analysis for quantifying and characterizing the type of speech children engage in as they solve a problem. In their approach, student verbalizations were viewed as a window on their thought processes. Chi [37] discusses the need for quantitative analysis of "messy" data in the learning sciences. Here, "messy" data is referred to as "verbal explanations, observations, and videotaping, as well as gestures. One reason for the need to collect this kind of data is the trend toward studying complex activities in practice or in the context in which they occur." [37, p. 271]. Our research focuses on studying a complex activity in practice: solving an open-ended engineering challenge. Through the coding and quantification of student activity captured in this "messy" data set, we created multi-faceted profiles of student approaches to solving the problem, which then allowed us to compare performances across students.

Twelve elementary students (six second graders and six sixth graders) took part in the study. The children worked individually to solve an open-ended engineering challenge based on age-appropriate LEGO robotics kits and craft materials. These consisted of LEGO WeDo and LEGO NXT Mindstorms kits respectively, with additional parts from resource kits and craft materials such as paper, blocks, and markers. All the students started with the robotics curriculum [11] in Kindergarten, which uses a mediated learning approach [38] combining teacher instruction, structured activities, and open-ended engineering design challenges. All second and sixth grade research students have been at the school since kindergarten so they had robotics for three and seven years respectively. While we originally theorized that there would be clear developmental differences between grade two and grade six students (and perhaps gender differences due to cultural pressures in older girls), a preliminary analysis did not support these theories and these results will be the subject of a future paper.

Participants were six grade two and six grade six students. These students were typically developing and were chosen for their ability to verbalize their actions to the researcher. Six students identified as girls and six students identified as boys. There were 3 boys and 3 girls from each grade. The school is a small, rural elementary school (PK-6) located in Western Massachusetts with 158 students. The school is 94.9% white, 19% of students have identified disabilities, 1.9% are English language learners, 25% are classified as low income [39].

3.2 Data Collection and Types of Data

Students were videotaped to capture their discourse and building/programming moves. Through a talk-aloud protocol [40] combined with semi-structured clinical interview [41, 42] their verbal discourse was captured. Participants were gently reminded to talk-aloud if they lapsed into silence. For example, "Researcher: You've decided to split that middle piece into two parts. What was your thinking there?" The goal was to neutrally ascertain students' thought processes. The discourse, in combination with the videotape of the building and programming moves, comprised the main data for this study. The use of "careful observation of the child's work with 'concrete' intellectual objects" [42, p. ix] was critical to later analysis of the building of the engineering prototypes.

The open-ended design challenge given to the students was to create a safe and fun amusement park ride. Before students built their amusement park ride for the main part of the research, they did a warm-up task. After the warm up task was completed, students did the main task of independently creating a **safe** and **interesting** amusement park ride using the provided, age-appropriate LEGO robotics kits, and craft materials. See Appendix E for the actual research prompt. Hard copy readers can go to https://kidsengineer.com/?page_id=1836.

For both tasks, participants were videotaped from the side. The first author took field notes during the sessions. Other data that helped characterize the designs and triangulate the video data were captured including: elapsed time of design activity, design artifacts, photos of the design in progress and completed, and the computer program developed (if any). In summary, the raw data for this study consisted of video of the student-built designs and a post interview, field notes, photos of in-progress and completed warm up and main task builds, and their finished computer program.

3.3 Ride Rating Rubric

The student-built designs were first analyzed using a ride rating rubric (Appendix A). The ride rating rubric considers two elements: (1) the functionality of the ride, and (2) the ride originality. These elements correspond directly to the requirements from the prompt that the ride be "safe" and "interesting". The originality aspect refers to the degree to which a child is creating a design based on their own unique ideas. The functionality aspect takes into account how well the student met the requirements as well as the stability, scale, and symmetry of the ride. These two elements are judged on a scale of 1 to 4, then averaged. In addition to the initial ride rating analysis, the raw video and transcript data were transformed into derived data that could be further analyzed. The derived data was classified into engineering design process (EDP) data and key factors rubric data.

3.4 EDP Data

To analyze the engineering design process (EDP) data, we developed a model for the different phases of the EDP (discussed above) shown in Fig. 2.

The specific phases in our EDP models are: problem definition, planning, researching, building, rebuilding, programming, reprogramming, evaluating, and sharing out. Each phase is defined as follows:

PLAN – subject was planning some aspect of their design, typically verbally.

RESEARCH – researching a problem or possible solution. Looking for parts was considering research if it affected major design decisions before building started. Otherwise, it was considered part of building.

BUILD/REBUILD – normal building or rebuilding, which includes looking for parts unless the looking for parts was part of researching the feasibility of a potential design.

PROGRAM/RE-PROGRAM – programming or reprogramming the robot.

EVALUATE – evaluate by testing the prototype physically, by visual inspection, or by evaluating the whole system typically by running the program.

There were two EDP codes that were not planned but were added. SHARE-OUT and PROBLEM-SCOPING (for the full EDP Code Book, see Appendix B). Problem scoping is defined by Atman et al. (2008) "as the stage of the design process during which designers explore the relevant issues and set the boundaries of the problem they will continue to solve" (p. 235), and is presented as problem definition in our model. We basically defined PROBLEM-SCOPING as asking clarifying questions about the design problem. The problem, although open-ended, was very well defined so there were a very small number of problem scoping instances observed. These instances were coded but not analyzed since they were so few in number (nine short instances in six subjects). It was anticipated that the SHARE-OUT of the project

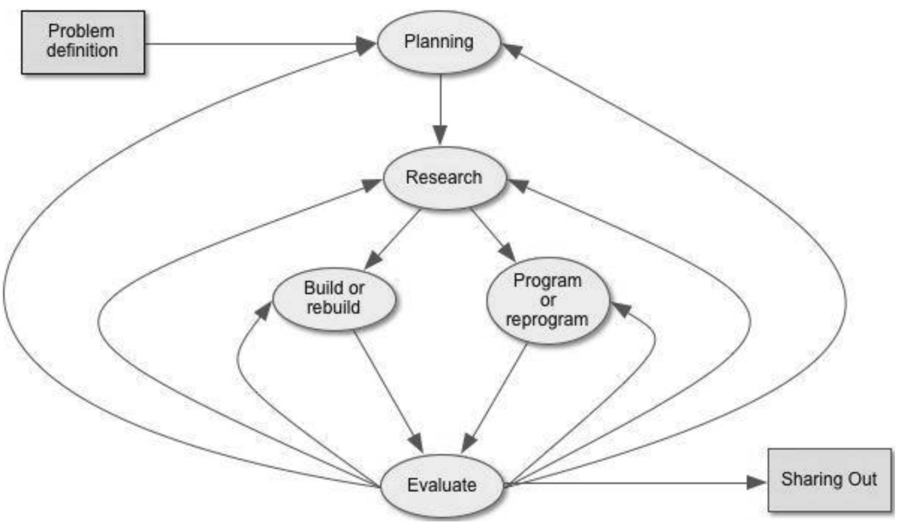

Fig. 2. Engineering design process model for study.

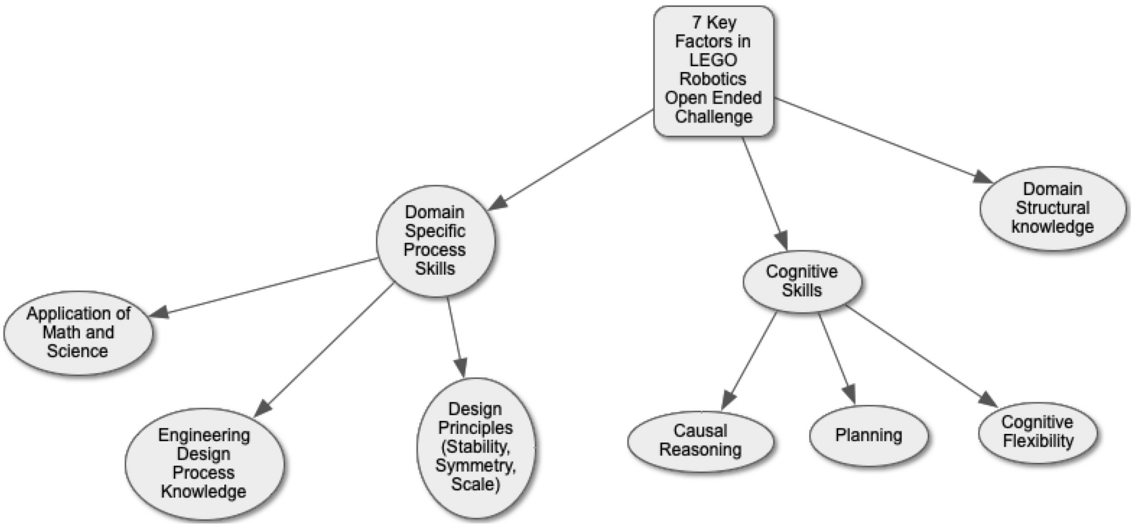

Fig. 3. Seven key factors in LEGO Robotics Open Ended Challenge.

would occur in the post-make (after the build) interview and would not be coded. However, a few students did significant, unprompted sharing out in the form of post-make drawing so these instances were coded and analyzed.

3.5 Key Factors Rubric Data

The other type of derived data is the key factors rubric data. The key factors rubric (Appendix C) focused on what we theorized (and later verified) were the most significant aspects of the student activity that occurred. The key factors were needed because our original hunch that development (that is, age) would explain EDP timeline and ride rating differences did not seem to be the case (see Fig. 3). The key factors can be grouped into cognitive skills, EDP skills. and domain structural knowledge.

Along with the intended built complexity (explained in the next section), the seven key factors explained the differing EDP timelines and differing ride ratings we observed in the subjects. The seven key factors were found theoretically from the literature, theorized from our own experience as robotics teacher and robotics researcher respectively, and from the research data. We noted important and repeated observations of phenomenon when viewing the video, reviewing notated transcripts, and considering the final model rides. We observed the apparent importance of LEGO experience and the EDP process, we observed successful and less successful designs, and we included the most significant predicted and observed factors from the literature as verified by the analysis below. Table 1 summarizes each key factor: their category, origin, and definition.

In summary, our key factors rubric included three types: (1) domain specific process skills related to knowledge of engineering and design – application of math and/or science in problem solving,

Table 1. Key Factor Summary Table

Key Factor	Type	Origin	Definition/More info
Application of math and science	Engineering domain specific process skills	Literature/experience	Ability to apply math and science to engineering problem
EDP knowledge		Literature/experience	Knowledge of and utilization of the engineering design process
Design principles		Emerged from study data	Rules of thumb. In the LEGO robotics context, stability, symmetry, and scale
Causal reasoning	Cognitive skills	Literature	Ability to see cause and effect. Consists of predictions, implications, inferences, and explanations
Planning		Literature	Formulation of a program of action to achieve an end
Cognitive flexibility		Literature	Ability to consider multiple bits of information or ideas at one time and actively switch between them when engaging in a task
Structural Knowledge	NA	Emerged from study data	Integrated schema of domain knowledge

knowledge of the engineering design process, and knowledge of design principles of scale, symmetry, and stability; (2) students' use of cognitive skills – cognitive flexibility, planning, causal reasoning; and (3) LEGO structural knowledge (e.g., knowledge of pieces, connecting techniques, etc.).

3.6 Build Complexity

Intended build complexity emerged as an important factor that had not been coded explicitly but had a basis in the theoretical framework, previous research, and our experience as robotics teacher and robotics researcher. In the context of open-ended design problems such as the amusement park ride task studied here, students choose what they wanted to build, which defined the intended build complexity. We include the word intended to reflect the complexity of the ride students intended and not just a teacher post facto judgement of how complex the ride turned out to be. We made a judgement by analyzing the video to determine if there was a significant difference between the stated intended ride plan and the actual ride. In seven subjects in this study, there was not. However, one student was not able to build her intended ride so her intended ride was much different in complexity from her actual ride. That case will be discussed in the results section.

According to Funke [43] and Jonassen [26], the most relevant aspects of problem (or build) complexity are the structuredness of the problem, the number of issues, functions, or variables in the problem, and the degree of connectivity between the variables. The ride challenge and robotics in general, depending on what the student chooses to build, can be high complexity since they are ill structured, have a high number of variables, functions, and issues, and can have connectivity between the variables. The build complexity rubric defines levels of complexity, based on these definitions and the demands of the task itself. See Appendix D for these defined levels.

3.7 Data Preparation, Coding and Interrater Reliability

Eight and a half hours of video sessions were transcribed, time stamped, and the verbal output from the talk aloud and clinical interview protocols [44, 45] was segmented. The purpose of segmenting is, "to break the verbal text into units (or segments) that can be coded with a pre-defined coding scheme" [45, p. 332]. Verbal output was generally easy to segment because it consisted of short question and answer snippets. In this study, there are two different "tracks" of data: verbal and physical. Note that similar studies only look at the verbal output of participants who work in teams [45, 46]. Talk was segmented when there was a change of speaker. For longer participant text in a transcription, talk was broken into additional segments by long pauses (more than 2 seconds) or clear changes of topic. Verbal segments were also split into multiple segments during the coding process if there was an EDP phase transition detected in the middle of a segment.

Because this study was interested in comparing individuals and because the physical building is so important to LEGO robotics, the physical building and programming activity of each participant were segmented by the first author with assistance from a graduate student. By examining the building moves of a number of participants, a unique physical move segmenting scheme was developed. Physical activity descriptors were defined to have a similar level of atomicity. The lower-level physical activity descriptors ultimately allowed interpretive coding of EDP phase transitions in combination with verbal output segments. Therefore, transitions between EDP phases were determined by both the students' physical building moves and their verbal output. For example, if the student stopped building with the LEGO parts and moved their design to see if it worked, it was clear that a transition from building to evaluation had occurred.

Physical activity was transcribed by activity descriptors such as pointing, gesturing, searching (for parts), connecting (parts), and moving. When the physical activity changed, a new timestamp and descriptor was inserted.

A fully time-stamped and segmented extract of a transcript is shown below.

[00:07:14] {connecting} Girl 05: I think this is going to be the last layer, and then I'm going to put the base through the middle.
[00:07:18] {searching}
[00:07:19] {connecting}
[00:07:23] {moving}
[00:07:24] Girl 05: Wait a second. (Lifts structure)
Researcher: What did you notice?
[00:07:29] Girl 05: It's uneven.

It became obvious very early that the study would have to account for the frequent occurrence of overlapping and different verbal and physical EDP phases. For example, a student could be building and talking about their plan for what comes next at the same time. To address this issue, we chose to represent each phase independently and exactly capture the overlapping phases. From the data, we created custom error X-axis bars that show the duration of the phase and any overlap with another phase, an example of which is presented in Fig. 4. Note the overlapping share and research phases that start around 0:02:13.

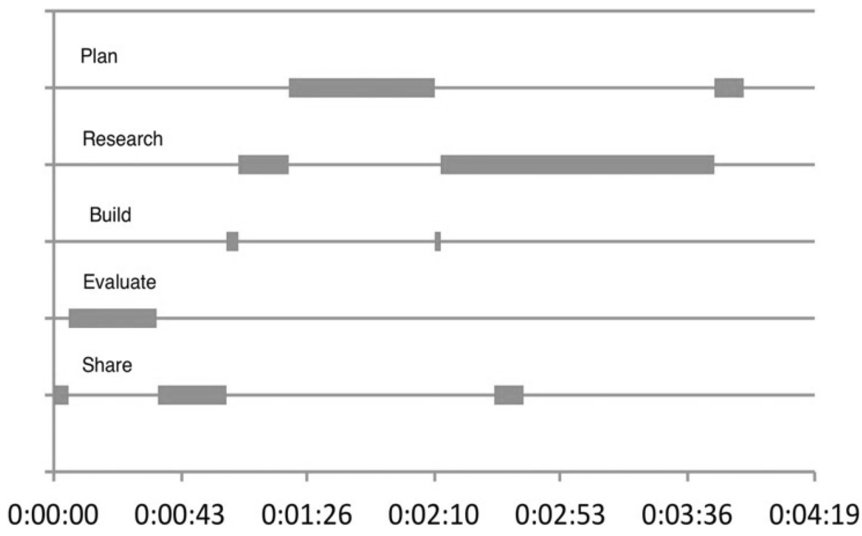

Fig. 4. Sample EDP timeline.

Multiple coding passes were made to ensure consistent and complete application of EDP codes and the various rubrics used in this study. A second coder was used to refine the coding dictionary and rubrics. After an initial training period, twenty percent of the EDP data and the factors rubric data were coded by the first author and a second coder, over 80% (83.3% and 87.13% respectively) intercoder reliability was achieved using Krippendorff's alpha [47, 48]. The 80% threshold was the same (or better than) similar studies with college level engineering students [49]. For the ride rating rubric which requires robotics teaching experience to holistically evaluate, 63% reliability was achieved. There were no systemic rating errors in any specific direction on the ride rating, so the preliminary results do not appear to be in error. A total of 312 pages of coded transcripts were produced.

See below for an extract of a coded transcript. The EDP codes (in square brackets) were placed directly after the timestamps and the factors EDP-related notes were placed at the end of each segment for clarity. Simultaneous EDP phases were indicated with a "2:" before the EDP phase code. The building moves, as well as the discourse, were transcribed and inserted using curly brackets immediately after the timestamps.

[00:17:01] {connecting} [BUILD-REBUILD]
[00:17:03] [2:EVALUATE-VERBAL] Boy 05: I put this on backwards.
[00:17:04] {moving} [EVALUATE-PHYSICAL]
[00:17:05] [2:END]
[00:17:07] [BUILD-REBUILD]
[00:17:14] {moving}
[00:17:15] {connecting} Researcher: I notice you're keeping your design so that it's usually the same thing on the other side, all the time.
[00:17:32] {gesturing} [EVALUATE-VERBAL] Boy 05: Yeah, so it's not off balance. If it's off balance, it has more likely to tip over.

Two programs were developed in the Python programming language [50] to extract the timestamps and codes from the transcripts. The two programs were a code scanner and a code extractor. The code scanner checked for valid codes and common errors. When coding errors were detected in either program, they were corrected and rechecked. The error detection improved the validity of the data and the reliability of the results. The main code files were then imported into Microsoft Excel. Once the data were imported into Microsoft Excel, a number of different types of visualizations were produced including EDP timelines presenting in the results section.

3.8 Data Analysis

The analysis looked at the frequency and distribution of events in the EDP timelines of the twelve students. This methodology is called inductive contrastive analysis [51]. Patterns were searched for in the EDP timelines of students. A similar approach has been used in studies of the design process of undergraduate students [47, 51, 54] and in novice/expert engineering studies [18]. As an example, Atman[52] found what was considered an ideal EDP timeline shape by looking at EDP timelines of expert practitioners and undergraduate engineering students.

The first step was to again look for patterns in EDP timelines of the students by the key factors such as domain structural knowledge, causal rea-

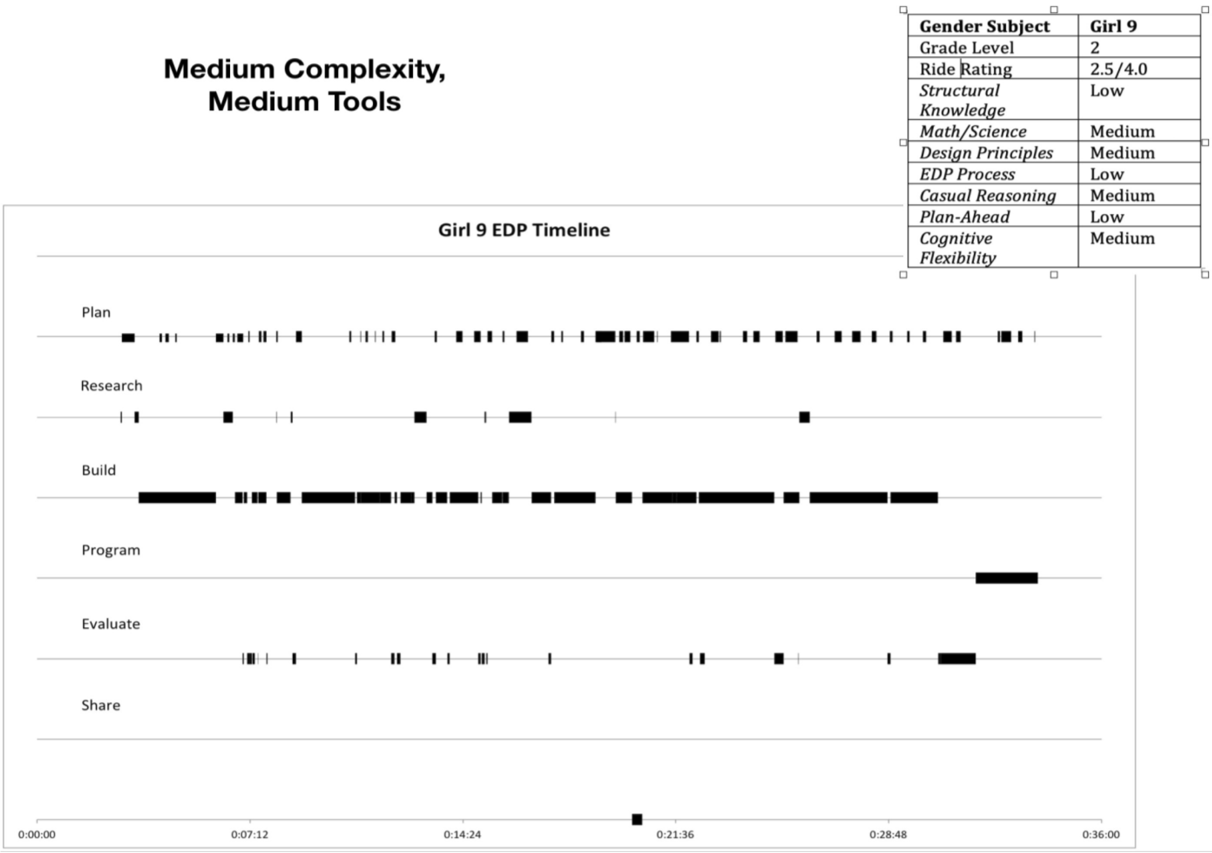

Fig. 5. EDP timeline summary.

soning (CR), cognitive flexibility (CF), and quantified in the key factors rubric (Appendix C). The EDP timelines were labeled with the key factors described above and sorted by each key factor in turn to see if patterns emerged. An example of the visualization created for this purpose is shown in Fig. 5.

When a pattern did emerge, the key factors rubric (Appendix C) was used to help quantify these factors for each design and design process, and appropriate visualizations (shown next) were created to show the relationships between the key factors and ride ratings.

4. Presentation and Discussion

4.1 Key Factors

Each of the 12 students' video recorded design sessions were analyzed using the ride rating rubric and the key factors rubric. In Table 2, we present the results for each student along all of the elements on the key factors rubric and the ride rating. We

Table 2. Factor ratings. * = mix of high and low ratings.

		EDP Factors			Cognitive Skills					
Subject	Structural Knowledge	Math/ Science	Design Principles	EDP Process	CR	Planning	CF	Tools	Build Complexity	Ride Rating
Boy 06	Medium	Low	Low	High	High	Low	High	Medium	High	3
Boy 07	Medium	Low	Medium	Medium	Medium	Low	Low	Medium	Medium	3.5
Boy 08	Low	High	Low	High	Low	High	Low	Low*	Medium	2
Girl 06	Low	Low	Medium	Medium	Low	Low	Medium	Low	Low	1.5
Girl 08	High	High	High	High	High	High	Medium	High	Low	3
Girl 09	Low	Medium	Medium	Low	Medium	Low	Medium	Medium	Medium	2.5
Boy 03	Low	Low	Low	Low	Medium	Low	Medium	Low	Low	2
Boy 04	High	Medium	High	Medium	High	Low	Medium	Medium	Low	2.5
Boy 05	High	Medium	High	Medium	High	High	Medium	High	High	3.5
Girl 03	Low	Low	Low	Medium	Low	Low	Low	Low	High	1
Girl 04	Low	Low	Medium	Medium	High	Medium	Medium	Medium	Medium	3
Girl 05	High	High	High	High	High	High	High	High	High	3.5

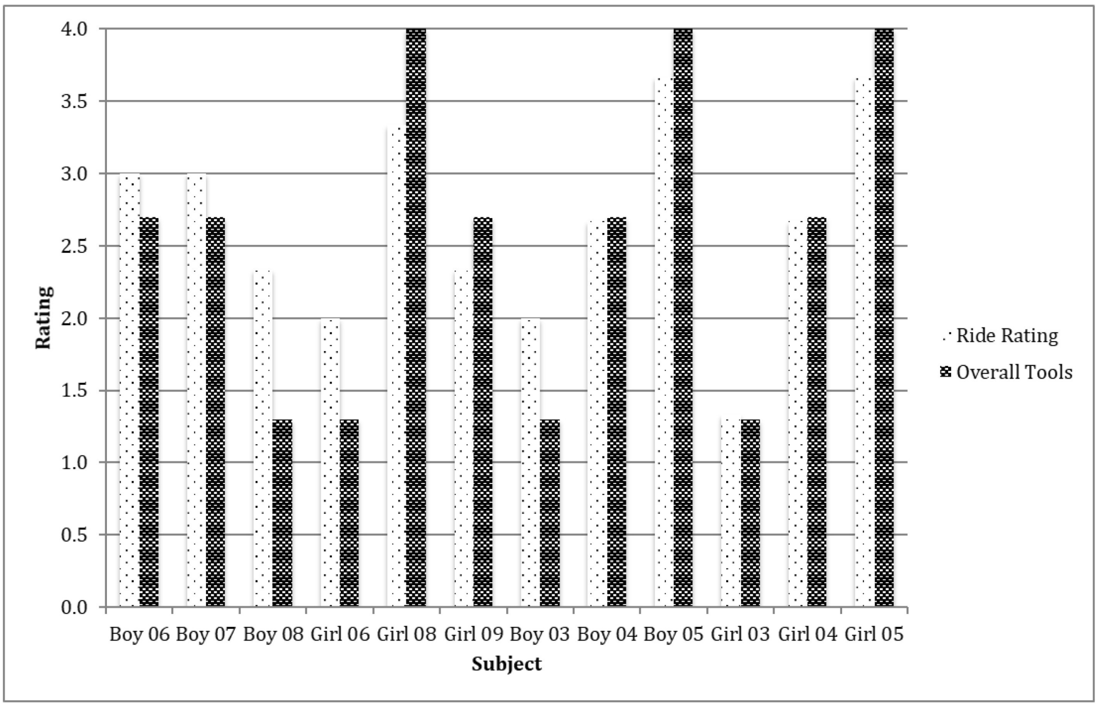

Fig. 6. Final design ride rating versus tools. Note: graph from left to right shows second graders to sixth graders.

averaged the overall rating scores on the key factors rubric and created a combined ranking under an overall category we have called Tools. Tools refers to the overall tools each student brought to the task, which turned out to be a mix of structural knowledge, engineering design process factors, and cognitive skills. Next, we provide the build complexity and the overall ride rating for each students' design.

We now present the individual key factor ratings by overall ride rating to establish the importance of each and discuss each key factor in turn.

4.1.1 Key Factors by Ride Rating

We begin by reviewing the overall tools rating. There appears to be a relationship between the overall tools the students brought to the task and the final design ride rating (see Fig. 6), which shows the validity of the seven key factors in explaining the overall ability of students to realize their design ideas. We observed that the greater the use of overall tools in designing, building, and testing the ride, the better the ride rating. Note that the key factor ratings were converted to numbers for direct comparison to the four-point ride rating as follows: high (4) medium (2.7), and low (1.3).

Although the sample sizes in this study were not large enough for statistical analysis, graphs were used to support the validity and reliability of observed relationships. In other words, the graphs supported our ideas on what we observed (along with other qualitative methods such as video analysis, rubric ratings, and reviewing of field notes). Let's look at each key factor individually.

4.1.2 LEGO Structural Knowledge

Domain specific knowledge about LEGO connection techniques emerged from the study as a key factor in the ability of students to realize their design ideas especially as the build complexity increased in some of the grade six designs. Recall that structural knowledge refers to a well-integrated and organized knowledge base in a specific domain. While the relationship to final ride rating does not appear to be as strong as the overall tools rating, it still appears to be a significant factor (see Fig. 7). Some students who had low structural knowledge compensated with other strengths (Boy 8 and Girl 9, for example).

Boy 5 and Girl 5 had extensive LEGO connector knowledge and also possessed meta knowledge about how the various LEGO connection techniques were related to each other. This structural and domain knowledge helped them design their highly rated and highly complex rides. Some of the students needed more structural knowledge to be successful. While the curriculum used in this study [11] identifies key WeDo connector parts, additional work is needed to map connection techniques and when to use them. For example, many students in this study lacked the knowledge that to make an axle move a beam, a cross to cross connection is needed. (Interested readers can refer to the

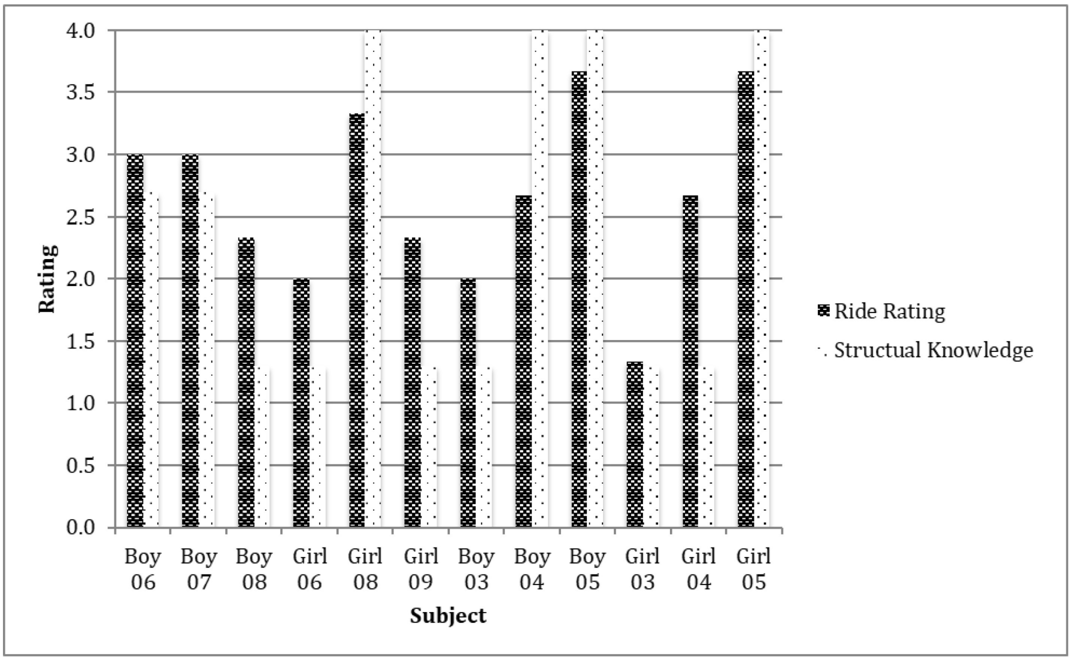

Fig. 7. Final design ride rating versus LEGO structural knowledge. Note: graph from left to right shows second graders to sixth graders.

bottom middle LEGO part of eFigure 2 (see https://kidsengineer.com/?page_id=1836 for all eFigures)). Once connector pairings are mapped to their functions, activities need to be developed to help students understand which connectors might work – domain knowledge – and also gain structural knowledge of the relationship between the different connectors.

In addition to structural knowledge, three engineering design process skills emerged as critical to the development of a quality design for students in this study: the application of mathematics and science to engineering, the application of the design principles of symmetry, scale, and stability, and the engineering design process knowledge.

4.1.3 Application of Mathematics and Science

Some builders were able to apply mathematics and science to their designs successfully, which is commonly defined as being integral to engineering [2]. We found that the correlation with ride rating is not quite as strong as the other factors (interested readers can refer to eFigure 3). This makes sense as not all rides at the elementary level require the application of mathematics or science to a significant degree. However, it remains an integral aspect of engineering to teach to younger students [2, 18].

Girl 5, the strongest builder in the study, did apply mathematics and science to her design both in counting beam holes to find the middle and in using gearing up. She did need some teacher scaffolding, in the form of a neutral question, to figure out how to apply between gearing up. The first author had to re-state Girl 5's previous statement about gearing up (namely, that the smaller gear rotates faster) before she could correctly apply science knowledge to her design. This is consistent with other research that teacher scaffolding is needed to help students apply science in design problems [18, 53, 54].

4.1.4 Design Principles

Students with the highest rated designs and design processes attended to and understood certain design principles: stability, symmetry and, to a lesser extent, scale (interested readers can refer to eFigure 4). There is a high correlation between ride rating and the use of design principles. One exception is Boy 6, who chose a low complexity build despite having some key strengths (such as writing a complex program). His particular build did not require design principles to work.

Students such as Girl 8 frequently cited and applied these design principles, while other builders who had difficulty realizing their design ideas such as Girl 3 did not. This indicates that learning activities should be created that teach these principles to students, especially symmetry and stability. These activities should include the structural knowledge that symmetrical structures tend to be stable, as evidenced by the use of connecting beams in multiple places. Boy 4 was able to articulate concerns about stability and symmetry explicitly, in both cases, showing use of engineering design principles. (Note that symmetry typically results in more stable structures, so the two design principles are related.)

[00:10:50] Boy 04: I'm going to do it to both sides again.
Researcher: It looks like everything is the same on both sides with your design. Why do you do that?
[00:11:02] Boy 04: So, it's not like . . . if there is a car going on it, it could turn one way instead of the other, but I wanted it to be one way. It just works better, I think.
[00:27:45] Boy 04: I'll just put this on this one too, so it's a little more stable.

4.1.5 EDP Process Skills

Most students had good knowledge of the engineering design process itself. 10 were rated medium or high. (Interested readers can refer to eFigure 5). Presumably this came from their exposure to the engineering design process due to yearly robotics units starting in kindergarten. In some cases, having a strong EDP compensated for less reliance on cognitive skills. Both Boy 8 and Girl 3 are examples of this. Students with advanced EDP skills exhibited subskills such as: systemic testing (Girl 5), control of variables (Girl 5), troubleshooting tactics (Girl 4, Girl 5, and Boy 5) and, in general, a good balance of time spent in different EDP phases, most notably upfront planning and research (Boy 5, Boy 8, and Girl 8). While students showed good EDP overall, instructions in specific techniques such as control of variables or domain specific troubleshooting tactics would further benefit students.

4.1.6 Causal Reasoning

Cognitive skills emerged as playing a key role in this study. As we hypothesized, causal reasoning - in the form of predicting the effects of design decisions – and inference – in the form of inferring what went wrong when testing – were key factors in the EDP of students (interested readers see eFigure 6). The most successful and advanced builders (Girl 5, Boy 5, and Girl 8, for example) had strong causal reasoning skills as measured by the key factors rubric. Skill in prediction increased the likelihood of making productive design decisions more often than students with less developed causal reasoning skills. Good inference skills allowed faster determination of non-productive design decisions so they could be corrected. Boy 5 had very good prediction skills. In this example, he decides to use a gear piece as a connector to both hold the seat assemblies together and turn the swings. He also predicts that an extension is needed so the rider would not hit the structure.

[00:44:31] Boy 05: I would need to add a gear.
Researcher: Oh, gear. What's the gear do?
[00:44:40] Boy 05: It would turn the swings.
[00:45:17] Boy 05: I think these are going to be the swings. I might have to add an extension, because when this turns, if the swings were here, the person would hit, or it might go just barely by. . .

His successful predictions saved him time and effort and also worked well. (Interested readers can see eFigure 7 for a picture of the gear and how it was used.)

We found that, in many cases, it was hard to determine if an incorrect prediction was a result of lack of structural knowledge or lack or causal reasoning or both. For example, if the motor is not connected to receive power, did the student not have the knowledge to understand that it needed to be connected or did they have the knowledge but did not have the causal reasoning abilities required to use that knowledge? One interesting example of this was Boy 8, who put the motor on the seat rather than on a tower type structure.

Boy 8 did not predict that the cord would become tangled even though it seemed obvious to the adult researcher. Note that if there is missing domain knowledge, there is no way that the student can create structural knowledge, which, by definition, integrates different domain knowledge. Again, this could also be interpreted as a lack of causal reasoning skills.

There is some evidence to suggest causal reasoning and the lack of structural knowledge are separate. Girl 4 scored high in causal reasoning and low in structural knowledge. Girl 9 scored medium in causal reasoning and low in structural knowledge. Girl 9, in particular, used good causal reasoning skills to compensate for low structural knowledge. As an example of low structural knowledge, she mixed up the names and functions of the hub and motor. However, she successfully was able to predict and plan various build moves as seen in the example below.

[00:12:00] Girl 09: I'm going to add a few more white pieces to make it go down a little, maybe another one of these.

It seems likely that lack of structural knowledge can appear to be lack of causal reasoning but that they may, in fact, be two different phenomena.

Causal reasoning is generally considered to be developmental (Fuson, 1976; Piaget & Inhelder, 1969) and there were more high causal reasoning sixth graders than second graders. Open-ended engineering problems appears to be a good activity type to help develop causal reasoning in the form of prediction and inference as long as students also have the required structural knowledge as a basis for causal reasoning.

4.1.7 Planning

Planning was another key factor in elementary student engineering though its importance depended on a number of other factors. (See eFigure 8 for the relationship between planning and the final ride rating for this study). Planning depends on causal reasoning, specifically, prediction [26]. Most students had a clear planning style, which can be described as either a serial (Boy 6, Boy 7, Girl 6, Girl 9, Boy 3, Boy 4, and Girl 3) or systems approach (Boy 8, Girl 8, Boy 5, and Girl 5). Girl 4 had elements of both styles of building.

At 4:36, Girl 3 clearly states the serial building approach.

Researcher: When you are thinking about your Ferris wheel do you plan just the first part and then worry about the rest later or do you have an idea in your head about what the whole thing is going to be?

[00:04:36] Girl 03: I usually just start with one thing and see how it goes.

In the case of Girl 3, who was unable to finish her ride, the lack of an overall plan before building caused her major problems getting her subassemblies connected later. Her ride, which is similar in concept to that of Boy 5 and Girl 5, would likely have been more successful with an overall system plan. Girl 5 and Boy 5 were able to successfully build the same ride concept as Girl 3 but had a clear plan of building a base, tower, rotating seat assembly, and seats in that order. However, there were many successful serial builders who chose less complex builds such as Boy 6 and Boy 7. The implication for teaching is that the teaching of planning will especially help students with complex designs and low causal reasoning skills. Note that having immediate access to the building materials (LEGO pieces) may encourage a more serial or tinkering approach as opposed to a more formal pencil and paper engineering planning and design approach typical of engineering processes research at the undergraduate level (Atman et al., 2007).

4.1.8 Cognitive Flexibility

Cognitive flexibility emerged, as predicted from research, as the final important cognitive skill in the study. We observed a relationship cognitive flexibility and the final ride rating with one exception. Two students – Boy 7 and Boy 8 had low cognitive flexibility but were able to compensate for it with strengths in other key process skills or structural knowledge. (See eFigure 9 for more detail.).

We think of cognitive flexibility in two ways: positive and negative. A positive cognitive flexibility in this context consisted of both being willing to start over when an idea is not working and in having many different ideas for a particular problem. A negative cognitive flexibility was defined as non-optimal persistence. For example, this manifested as repeated stability or other issues that the student would keep repairing without addressing the underlying cause. Boy 7, for example, continually tried to make his ride spin with non-solid LEGO connection so he had to hand start his ride. Teachers can aid students showing non-optimal persistence with encouragement to rethink what they are doing, or they may be lacking a specific piece of domain structural knowledge. For example, Boy 7 needed to know that an axle needs to be inserted into a cross piece to make a stable connection. Of course, positive persistence or positive cognitive flexibility should be encouraged.

4.2 EDP Timeline, Key Factors, and Build Complexity

Our initial analysis of the data indicated that the seven key factors in combination with build complexity might explain the very different EDP timelines.

Table 3 presents the students work plotted on a matrix that indicates the level of complexity and the level of tools they used in creating their design. In the following section, we present different EDP timeline and where the individual student lies along this matrix. In so doing, we demonstrate the relationship of the engineering design process to the factors of interest in this study. Indeed, we argue that the shape and nature of the EDP timeline for each student was precisely defined by the complexity of the build chosen and the tools they brought to the task.

Let us begin by discussing the second grader Girl 8 who seemed to have high skills but chose a low build complexity (see Fig. 8). Girl 8 made a detailed and accurate drawing of her simple, non-motorized ride, which is shown as planning. She then built her ride. Research was not needed, nor was there very much evaluation. This made sense since she was

Table 3. Build complexity, and tools by EDP timeline shape. Note that the subjects' age (second graders bolded) do not display a clear age-related pattern

Build Complexity Tools	**Low**	**Medium**	**High**
Low	Boy 3, **Girl 6**	**Boy 8**	Girl 3
Medium	Boy 4	Girl 4, **Boy 7, Boy 6, Girl 9**	
High	**Girl 8**		Girl 5, Boy 5

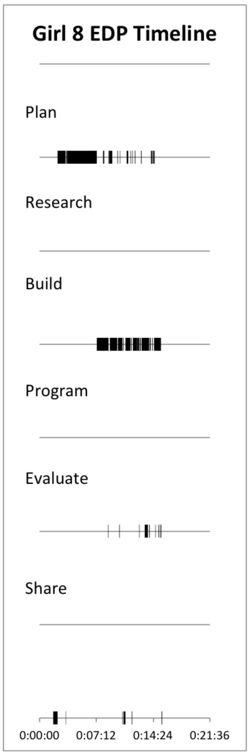

Fig. 8. Girl 8 EDP timeline. High tools, low complexity build.

very skilled but chose a very simple design; she could basically plan it out and build it without much iteration. In some sense, this might be considered an idealized EDP, where the plan works out exactly.

Sixth graders Boy 5 and Girl 5 had very similar EDP timelines (see Fig. 9). They both chose complex designs and brought high tools (skills) to the task. Their timelines show a long and dense mix of EDP phases with significant planning, research, and evaluation mixing in with the predominant build cycles. This shows a very productive EDP with a significant and meaningful utilization of each and every EDP phase.

Sixth grader Girl 3 tried (but could not finish) a ride idea very similar to Boy 5 and Boy 6 because she had low tools including low planning. You can see that she initially had a long period of building with no research but when her idea was not working, she then spent a long period of time trying to plan, research, build, and evaluate. However, she could never solve the technical problems she encountered. So, a student who tries to build a complex design idea without sufficient planning may not be able to complete their design. Note in the research setting, the researcher did not provide help to the student.

A typical, low tools, low complexity EDP timeline is shown in Fig. 11. While there is some planning, the sixth grader student (Boy 3) has a short, build-heavy EDP with little planning and evaluation. Such students have sufficient tools to realize their low complexity designs but will need help from teachers to gain more tools and more fully utilize the EDP to realize more complex design ideas.

Second grader Boy 6 showed a typical, medium

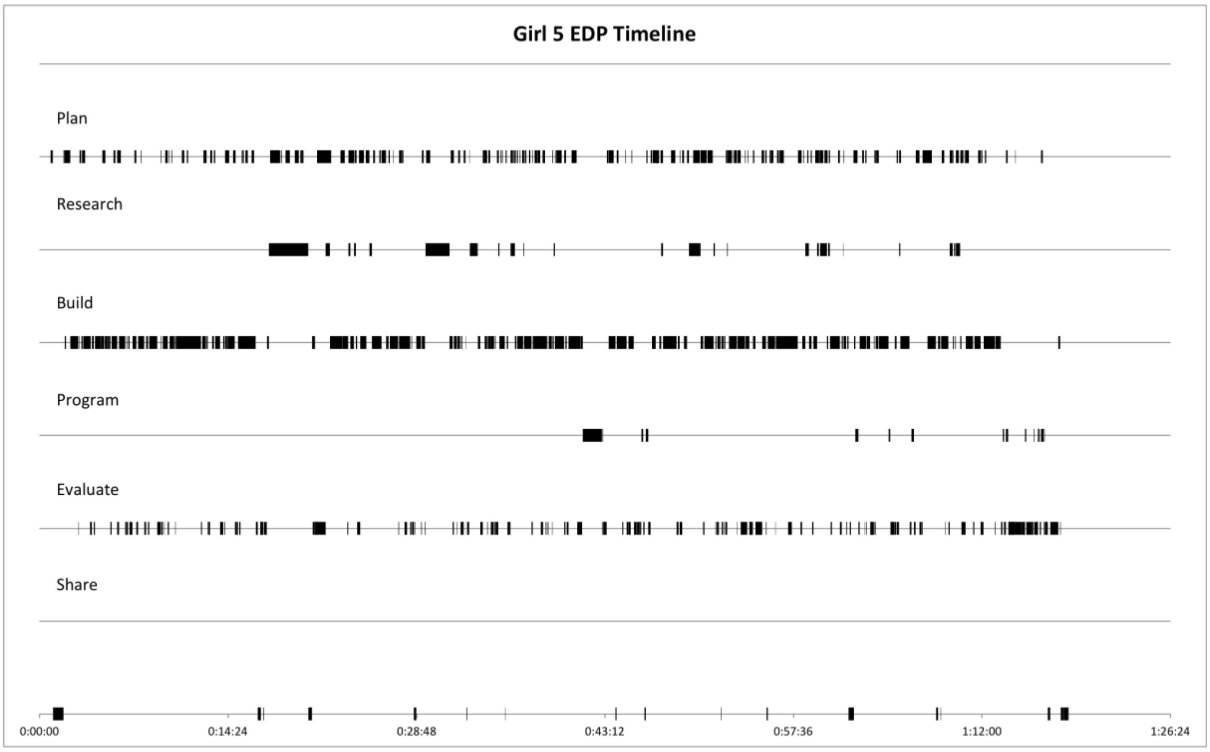

Fig. 9. Girl 5 EDP timeline. High tools, high complexity.

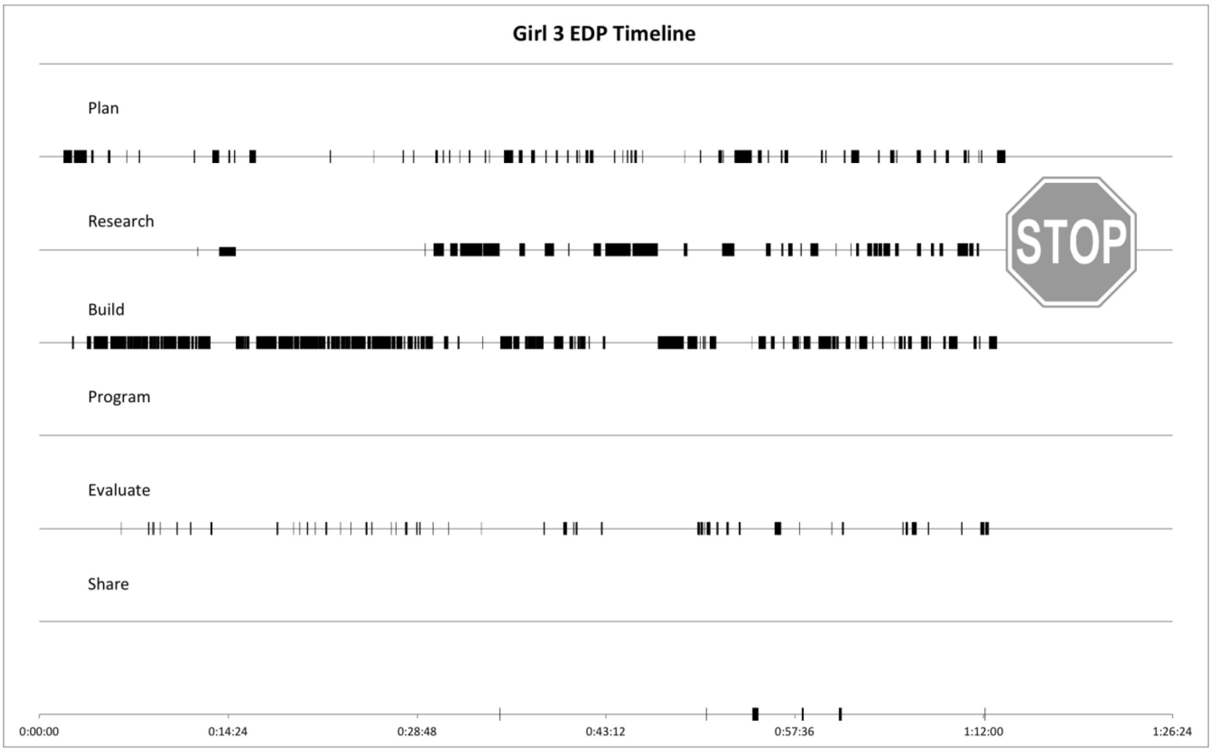

Fig. 10. Girl 3 EDP timeline. High complexity, low tools. The stop sign indicates that the build was stopped because the subject was stuck and could have continued indefinitely without making progress.

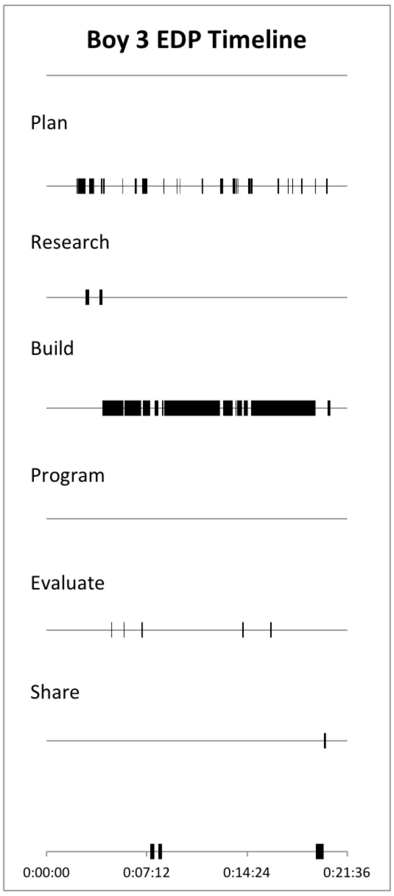

Fig. 11. Boy 3 EDP timeline. Low tools, low complexity.

tools, medium complexity EDP (see Fig. 12). Such students had shorter EDP timelines but did utilize all phases of the EDP. However, the design timelines show much less iteration and are more build heavy than the high tools, high complexity design timelines (See Fig. 11).

Exploring the relationship of these EDP timelines to students' tools and build complexity allows us to understand the relationship of our seven key factors to student learning in engineering design. We found that the seven key factors did make a significant difference in the final ride success and their EDP. Specifically, the students' structural knowledge of LEGO, their use of key cognitive skills critical to the engineering design process, their use of key domain specific engineering process skills, the complexity of the chosen design defined the quality of their final designs, and the shape of their EDP timelines. What emerged from the study were four different cases of tools and complexity (see Fig. 13). Students can be thought of as being in one of four quadrants. The arrow shows where we want students to go. We are not suggesting pushing students to highly complex designs for their own sake. Sometimes, simple designs are the best. We do have the goal that students be able to fully implement and express their own design ideas, whatever they are, using the provided materials.

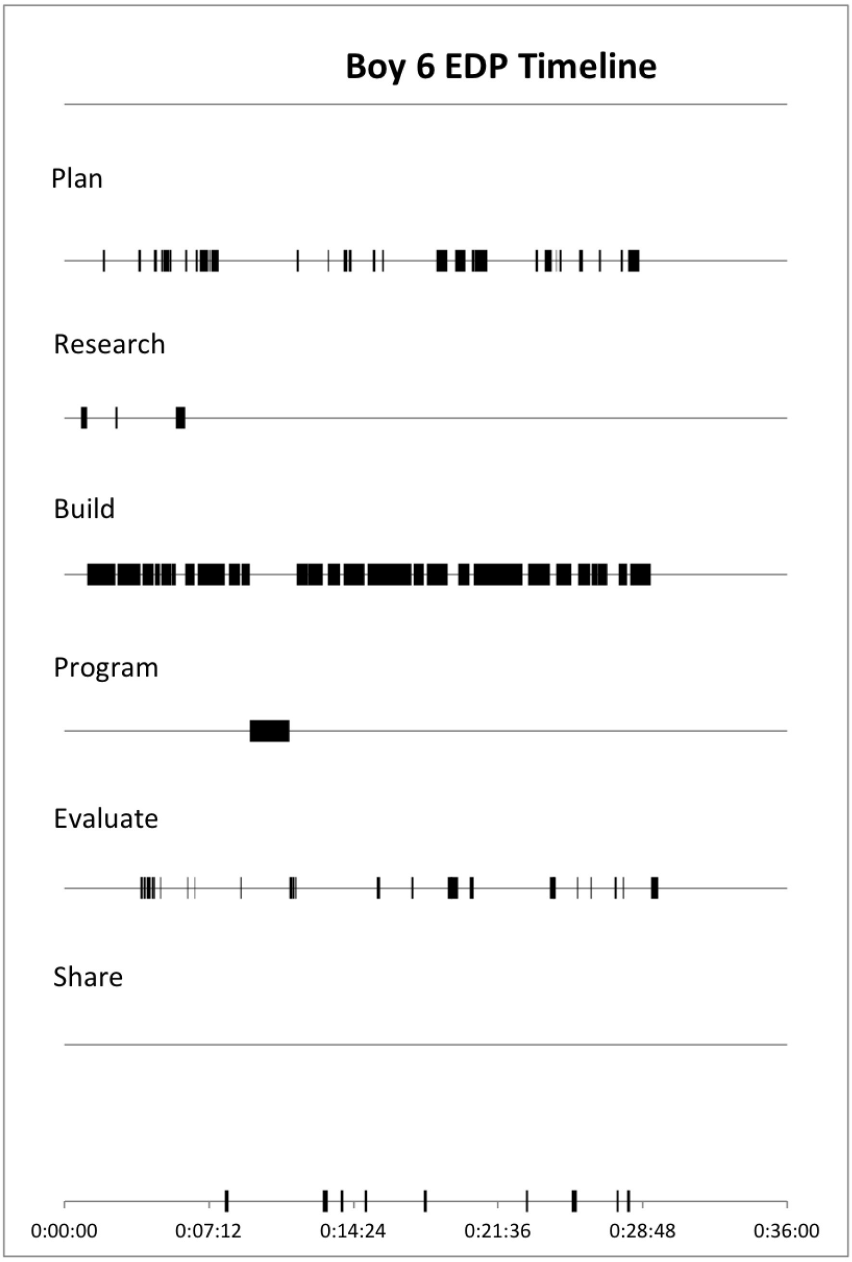

Fig. 12. Boy 6 EDP timeline. Medium complexity, medium tools.

- High tools, high complexity: some students built high complexity designs within their current high tool capabilities (Girl 5 and Boy 5). For these students with matching high tools and high build complexity, teachers should continue to move the students upward in the direction of the arrow by pushing for more complex builds by providing experiences and knowledge needed for continued growth along this continuum.
- Low Tools, Low Complexity – while students in this group (Boy 3 and Girl 6) have low tools, they also choose low complexity builds. While they succeeded in realizing their design ideas because their low complexity designed matched their low tools, they need scaffolding in domain specific process skills, cognitive skills, and/or structural knowledge of LEGO connection techniques in order to successfully build higher complexity designs.
- High tools, low complexity: some students with high tools built low complexity designs but could have done much more sophisticated designs (Girl 8, Boy 4). Requiring the use of more components such as a motor, computer, and sensor would have increased the complexity enough to make the task harder for students who have the requisite cognitive tools for higher complexity tasks. For these students, complexity should be high, or

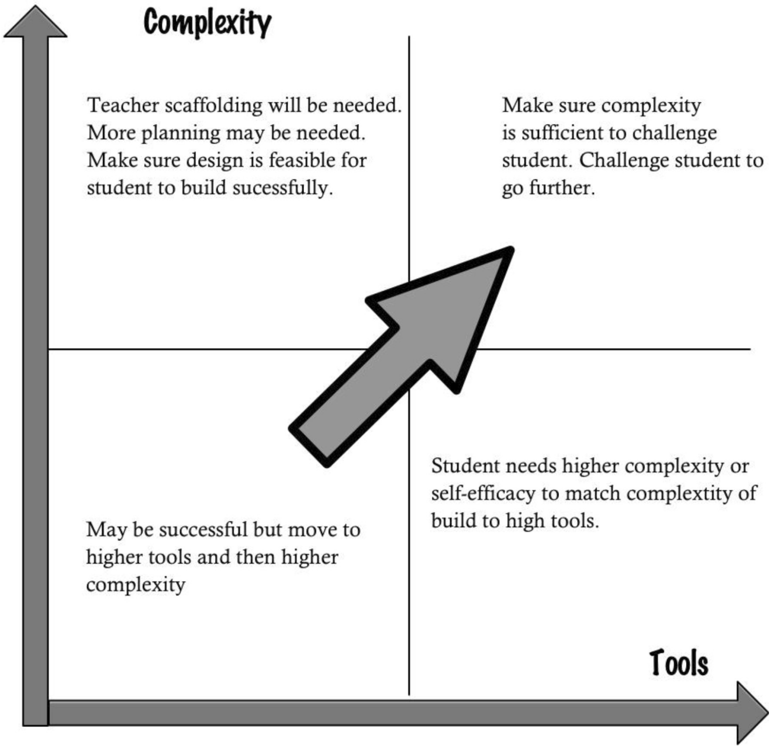

Fig. 13. Complexity and tools relationship.

the problem will be too easy, as we saw with Girl 8, who completed her ride without much failure or iteration, so she did not truly experience the full richness of the engineering design process.

- Low tools, high complexity – students in this group (Girl 3 and Boy 8) need scaffolding in LEGO connection and other domain specific skills in order to success building their high complexity designs. For Girl 3, additional scaffolding might have been: sketch out the overall design first, have her build the tower first, and provide direct instruction or scaffolding of additional LEGO connection methods. Structural knowledge will be improved just by doing LEGO engineering activities, but students will also need direct instruction. Many students compensated (Girl 9, Boy 8, and Girl 4, for example) for low structural knowledge with other strengths such as high EDP or strong causal reasoning or cognitive flexibility. In general, we do not recommend suggesting an easier design idea for students in this group unless their idea is completely impractical. That way, the student can feel empowered to realize their design ideas albeit with teacher help.

4.3 Limitations of Study

The coding of the engineering design process of students is an approximation and it is not possible to be 100% accurate because some building and verbal moves could be interpreted in different ways. However, the intercoder reliability results showed consistent interpretation across multiple students. Also, students did not always verbalize their thinking perfectly. The use of the dual physical track helped to ameliorate this limitation.

The small sample size of twelve was also a limitation of this study in terms of being able to generalize the results. However, the time involved to segment, code, and process the video was already substantial and is a limitation of this kind of research [45]. It was also a challenge to find qualifying, typical students at the small rural school and the makeup of the students in this small, rural public school was not typical of many public schools.

5. Conclusion

Elementary students' engineering design processes (EDP) were defined by build complexity and the overall cognitive tools that students brought to the task. These tools were found to be structural knowledge of LEGO and a combination of cognitive skills (casual reasoning, planning ability, and cognitive flexibility) and domain specific process skills (EDP process knowledge, application of design principles of stability, symmetry, and scale, and application of mathematics and science). Note that three of these - structural knowledge, EDP process knowledge, and design principles – were found in the literature review as being utilized by experts. Since these particular factors did not appear to be developmental, this suggests that they could be taught to students explicitly. Additional research is needed to determine more accurately the relative importance of the different factors. Future research studies may choose to focus on analyzing the efficacy of particular curricular and pedagogical approaches to supporting student learning in each of the three types of approaches identified here: high complexity/low tools; matching complexity and tools; and low complexity/high tools.

In our experience, elementary engineering based on LEGO robotics in a K-6 yearly program shows rich affordances to develop student engineering and cognitive skills. Indeed, the results found in this paper seem to extend to middle and high school students by the first author, who now teaches middle and high school robotics. This study has provided significant characterization, insight, and implications for teaching elementary engineering to help sustain the natural interest and ability of all young children to design solutions to overcome the complex problems of today.

Appendices

See https://kidsengineer.com/?page_id=1836 for Appendices A to E.

References

1. F. R. Sullivan and J. Heffernan, Robotic Construction Kits as Computational Manipulatives for Learning in the STEM Disciplines, *J. Res. Technol. Educ.*, 2016.
2. S. Brophy, M. Portsmore, S. Klein and C. Rogers, Advancing Engineering Education in P-12 Classrooms, *J. Eng. Educ.*, **97**(3), Jul. 2008.
3. M. Gura, *Getting Started with LEGO Robotics: A Guide for K-12 Educators [Paperback]*. Eugene, OR: ISTE, 2011.
4. Next Generation Science Standards, 2012. http://www.nextgenscience.org/ (accessed Nov. 02, 2012).
5. M. Bers, *Blocks to robots: learning with technology in the early childhood classroom*. Teachers College Press, 2008.
6. M. Bers, L. Flannery, E. Kazakoff and A. Sullivan, Computational thinking and tinkering: Exploration of an early childhood robotics curriculum, *Comput. Educ.*, **72**, pp. 145–157, Mar. 2014.
7. E. Kazakoff and M. Bers, Programming in a robotics context in the kindergarten classroom: The impact on sequencing skills, *J. Educ. Multimed. Hypermedia*, **21**(4), pp. 371–391, 2012.
8. F. R. Sullivan, Robotics and science literacy: Thinking skills, science process skills and systems understanding, *J. Res. Sci. Teach.*, **45**(3), pp. 373–394, Mar. 2008.
9. F. R. Sullivan, The Creative Nature of Robotics Activity: Design and Problem Solving, in *Robotics in STEM Education*, Springer, 2017, pp. 213–230. Accessed: Sep. 16, 2017. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-57786-9_9
10. E. Cejka, C. Rogers and M. Portsmore, Kindergarten Robotics: Using Robotics to Motivate Math, Science, and Engineering Literacy in Elementary School, *Int. J. Eng. Educ.*, **22**(4), pp. 711–722, 2006.
11. J. Heffernan, *Elementary Engineering: Sustaining the Natural Engineering Instincts of Children*. Charlestown, SC: Printed by CreateSpace, 2013.
12. S. Papert, *Mindstorms: Children, Computers, And Powerful Ideas*, 2nd ed. Basic Books, 1993.
13. Merriam-Webster, *Engineering. In Merriam-Webster.com dictionary.* http://www.merriam-webster.com/dictionary/engineering (accessed Apr. 05, 2015).
14. D. Crismond and R. S. Adams, The informed design teaching and learning matrix, *J. Eng. Educ.*, **101**(4), pp. 738–797, 2012.
15. M. Portsmore, Scaffolding the Engineering Design Process for Elementary Students (PowerPoint Slides), 2011.
16. C. Roden, Young children's problem-solving in design and technology: towards a taxonomy of strategies, *J. Des. Technol. Educ.*, 2(1), 1997, Accessed: Oct. 07, 2013. [Online]. Available: https://jil.lboro.ac.uk/ojs/index.php/JDTE/article/view/375
17. C. Roden, How children's problem solving strategies develop at Key Stage 1, *J. Des. Technol. Educ.*, **4**(1), 1999, Accessed: Aug. 13, 2013. [Online]. Available: http://ojs.lboro.ac.uk/ojs/index.php/JDTE/article/view/404
18. D. Crismond, Learning and using science ideas when doing investigate-and-redesign tasks: A study of naive, novice, and expert designers doing constrained and scaffolded design work, *J. Res. Sci. Teach.*, **38**(7), pp. 791–820, 2001.
19. M. Welch, Analyzing the Tacit Strategies of Novice Designers, *Res. Sci. Technol. Educ.*, **17**(1), pp. 19–33, 1999.
20. R. Johnsey, Observing the way primary children design and make in the classroom: an analysis of the behaviours exhibited, 1993, Accessed: Apr. 20, 2015. [Online]. Available: https://dspace.lboro.ac.uk/dspace/handle/2134/1566
21. C. J. McRobbie, S. J. Stein and I. Ginns, Exploring designerly thinking of students as novice designers, *Res. Sci. Educ.*, **31**(1), pp. 91–116, 2001.
22. M. Portsmore, Exploring How Experience With Planning Impacts First Grade Students' Planning And Solutions To Engineering Design Problems (Doctoral Dissertation), 2009. [Online]. Available: https://www.researchgate.net/publication/252861164_Exploring_how_experience_with_planning_impacts_first_grade_students'_planning_and_solutions_to_engineering_design_problems
23. N. Cutting, I. A. Apperly, J. Chappell and S. R. Beck, The puzzling difficulty of tool innovation: Why can't children piece their knowledge together?, *J. Exp. Child Psychol.*, **125**, pp. 110–117, Sep. 2014.
24. N. Cutting, I. A. Apperly and S. R. Beck, Why do children lack the flexibility to innovate tools?, *J. Exp. Child Psychol.*, **109**(4), pp. 497–511, Aug. 2011.
25. W. Gardner and B. Rogoff, Children's deliberateness of planning according to task circumstances, *Dev. Psychol.*, **26**(3), p. 480, 1990.
26. D. H. Jonassen, Toward a design theory of problem solving, *Educ. Technol. Res. Dev.*, **48**(4), pp. 63–85, 2000.
27. S. T. Levy and D. Mioduser, Approaching Complexity Through Planful Play: Kindergarten Children's Strategies in Constructing an Autonomous Robot's Behavior, *Int. J. Comput. Math. Learn.*, **15**(1), pp. 21–43, Apr. 2010.

28. K. B. Cartwright, Insights From Cognitive Neuroscience: The Importance of Executive Function for Early Reading Development and Education, *Early Educ. Dev.*, **23**(1), pp. 24–36, Jan. 2012.
29. A. K. Stone-Macdonald, K. B. Wendell, A. Douglass, M. L. Love and M. L. Hyson, *Engaging Young Engineers: Teaching Problem Solving Skills Through STEM*. Baltimore, MD: Brookes Publishing, 2015.
30. R. J. Sternberg, Creative thinking in the classroom, *Scand. J. Educ. Res.*, **47**(3), pp. 325–338, 2003.
31. M. C. Davidson, D. Amso, L. C. Anderson and A. Diamond, Development of cognitive control and executive functions from 4 to 13 years: Evidence from manipulations of memory, inhibition, and task switching, *Neuropsychologia*, **44**(11), pp. 2037–2078, Jan. 2006.
32. M. D. Portsmore and B. M. Brizuela, AC 2011-1780: first grade students planning and artifact construction while working on an engineering design problem, presented at the *ASEE Annual Conference, Vancouver, BC, Canada, 2011*. Accessed: Feb. 21, 2013. [Online]. Available: http://jee.asee.org/file_server/papers/attachment/file/0001/1710/Draft_Portsmore_ASEE2011v2.pdf
33. A. Anning, Dilemmas and opportunities of a new curriculum: Design and technology with young children, *Int. J. Technol. Des. Educ.*, **4**(2), pp. 155–177, 1994.
34. M. Fleer, The science of technology: Young children working technologically, *Int. J. Technol. Des. Educ.*, **9**(3), pp. 269–291, 1999.
35. D. H. Jonassen and I. G. Ionas, Designing effective supports for causal reasoning, *Educ. Technol. Res. Dev.*, **56**(3), pp. 287–308, 2008.
36. A. Newell and H. A. Simon, *Human problem solving*, 104. Prentice-hall Englewood Cliffs, NJ, 1972.
37. M. T. H. Chi, Quantifying Qualitative Analyses of Verbal Data: A Practical Guide, *J. Learn. Sci.*, **6**(3), pp. 271–315, Jul. 1997.
38. J. Suomala and J. Alajaaski, Pupils' Problem-Solving Processes In A Complex Computerized Learning Environment, *J. Educ. Comput. Res.*, **26**(2), pp. 155–176, May 2002.
39. Massachusetts Department of Education School and District Profiles, *Massachusetts Department of Education School and District Profiles*. http://profiles.doe.mass.edu/ (accessed Mar. 11, 2015).
40. K. A. Ericsson and H. A. Simon, Verbal reports as data, *Psychol. Rev.*, **87**(3), p. 215, 1980.
41. M. E. Brenner, Interviewing in educational research, in *Handbook of complementary methods in education research*, pp. 357–370, 2006.
42. H. Ginsburg, *Entering the child's mind: The clinical interview in psychological research and practice*, Cambridge University Press, 1997.
43. J. Funke, Solving complex problems: Exploration and control of complex systems, *Complex Probl. Solving Princ. Mech.*, pp. 185–222, 1991.
44. K. A. Ericsson and H. A. Simon, *Protocol Analysis: Verbal Reports as Data*, Cambridge, MA: MIT Press, 1993.
45. C. J. Atman and K. M. Bursic, Verbal protocol analysis as a method to document engineering student design processes, *J. Eng. Educ.*, **87**(2), pp. 121–132, 1998.
46. M. E. McFarland and R. Bailey, Investigating Pattern in Design Performance of Interdisciplinary Undergraduate Engineering Student Teams, Jun. 2015. doi: 10.18260/p.24375.
47. D. G. Freelon, ReCal: Intercoder reliability calculation as a web service, *Int. J. Internet Sci.*, **5**(1), pp. 20–33, 2010.
48. K. Krippendorff, Computing Krippendorff's alpha reliability, *Dep. Pap. ASC*, p. 43, 2007.
49. C. J. Atman, M. E. Cardella, J. Turns and R. Adams, Comparing freshman and senior engineering design processes: an in-depth follow-up study, *Des. Stud.*, **26**(4), pp. 325–357, Jul. 2005.
50. M. Summerfield, *Programming in Python 3: a complete introduction to the Python language*, Addison-Wesley Professional, 2010.
51. R. Goldman, F. Erickson, J. Lemke and S. J. Derry, Selection in video, *Guidel. Video Res. Educ. Recomm. Expert Panel*, pp. 15–22, 2007.
52. C. J. Atman, R. S. Adams, M. E. Cardella, J. Turns, S. Mosborg and J. Saleem, Engineering design processes: A comparison of students and expert practitioners, *J. Eng. Educ.*, **96**(4), pp. 359–379, 2007.
53. D. Fortus, J. Krajcik, R. C. Dershimer, R. W. Marx and R. Mamlok-Naaman, Design-based science and real-world problem-solving, *Int. J. Sci. Educ.*, **27**(7), pp. 855–879, Jan. 2005.
54. S. Puntambekar and J. L. Kolodner, Distributed Scaffolding: Helping Students Learn Science from Design, *J. Res. Sci. Teach.*, **42**(2), pp. 185–217, 2005.

John Heffernan is currently Innovation Center Teacher at the Pioneer Valley Regional Schools. He designs and implements innovative engineering experiences. John is also Part Time Lecturer at Tufts Center for Engineering Education and Outreach. John's research interest is the engineering processes of K-12 students. John is a recipient of MASSCUE Pathfinder and Presidential Award for Excellence in the Teaching of Mathematics and Science awards.

Florence Sullivan is Professor and Department Chair, Teacher Education and Curriculum Studies at the University of Massachusetts, Amherst. Her research focuses on student collaborative learning with computational media. Specifically, she focuses on collaborative creativity, problem solving, group dynamics and aspects of computational thinking as they manifest while playing with robotics or other computational media. Her work has a special focus on understanding girls' experiences in these settings. A secondary research focus is learning in online environments including many of the same concerns: collaboration, gender and problem solving. She is also currently working on the development of a new research method that uses computational means to facilitate microgenetic analysis of discourse data; this work is funded by the National Science Foundation.