Studying Students' Understanding of Engineering Concepts through Their Sketches

Dr. Geoffrey L Herman, University of Illinois, Urbana-Champaign

Dr. Geoffrey L. Herman is a visiting assistant professor with the Illinois Foundry for Innovation in Engineering Education at the University of Illinois at Urbana-Champaign and a research assistant professor with the Department of Curriculum & Instruction. He earned his Ph.D. in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign as a Mavis Future Faculty Fellow and conducted postdoctoral research with Ruth Streveler in the School of Engineering Education at Purdue University. His research interests include creating systems for sustainable improvement in engineering education, promoting intrinsic motivation in the classroom, conceptual change and development in engineering students, and change in faculty beliefs about teaching and learning. He serves as the webmaster for the ASEE Educational Research and Methods Division.

Ms. Nicole Johnson, University of Illinois at Urbana-Champaign

Nicole received her B.S. in Engineering Physics at the Colorado School of Mines (CSM) in May 2013. She is currently working towards a PhD in Materials Science and Engineering at the University of Illinois at Urbana-Champaign (UIUC) under Professor Angus Rockett and Geoffrey Herman. Her research is a mixture between understanding defect behavior in solar cells and student learning in Materials Science. Outside of research she helps plan the Girls Learning About Materials (GLAM) summer camp for high school girls at UIUC.

Studying Students' Understanding of Engineering Concepts Through Their Sketches

1. Introduction

Sketches and other forms of graphical communication are central to both the practice and learning of engineering[1-4]. Visual representations play a critical role in helping students learn engineering concepts, socialize them into the engineering discipline, and facilitate or hinder the design process[5]. To help students practice and use these representations, several engineering colleges and classrooms have adopted tablet- and sketch-based instruction[6,7]. Despite the sustained interest in sketching on tablets and the importance of graphical communication and visual representations, our understanding of how students learn these representations and use them is poor[6,8].

In this paper, we present findings from an initial study on how students learn to use engineering sketches while solving canonical disciplinary problems.

2. Background

To better understand how students use and produce engineering sketches, we are conducting a series of novice-expert comparison studies. Investigations into the differences between experts and novices in their ability to process and recall information have provided a critical foundation in understanding how people learn[9]. Knowledge of these differences have led to the creation of foundational educational theories (e.g., ontological shifts[10,11]), assessment tools (e.g., concept inventories[12]), and research-based instructional practices suitable for the classroom (e.g., bridging analogies[13]) or computer-automated environments (e.g., hierarchical analysis tools[14]). When studying how people learn about and use scientific diagrams, these studies have revealed that experts and novices find different elements of diagrams salient and that they chunk visual information differently[9,15,16]. For example, experts emphasize underlying processes and functions in diagrams (e.g., focusing on the cycle of evaporation and rain illustrated in a diagram) while novices focus on surface features (e.g., focusing on the fact that there are clouds, lakes, and the sun in that same diagram)[16]. Critically, these studies have established that domain knowledge dramatically influences perception and understanding of visual representations.

Our study was guided by DiSessa's Knowledge in Pieces framework that argues that students' knowledge is fragmented lacking coherence across contexts. Novices construct their knowledge on demand, marshalling pieces of information that are cued by the context of the problem. Prior research on students' conceptual understanding of engineering concepts has revealed that students' ability to articulate their understanding is altered by perturbations in the visual presentation of the content[17,18]. For example, when students discussed shear stresses and strains in mechanics of materials, their reasoning was dependent on the physical orientation of the members undergoing axial loads or the physical orientation of stress elements drawn on the members[19-21]. It was revealed that students' reasoning overly relied on physical orientation, conceiving of "shear" as a *vertical* construct. Students would routinely replace the word "shear" with "vertical" during reasoning[19-21], but this linguistic shift is only viable in certain circumstances. Similarly, while students could readily access conceptions of shear stress to explain member failure and deformation when presented with schematic drawings, they struggled to access this same conceptual understanding when presented with pictures of real-life concrete cylinders under axial loads that failed at 45 degree angles[19-21].

In digital logic, the context-dependence of students' conceptual understanding was similarly revealed. When analyzing Boolean logic problems, students adopted dramatically different solution procedures and cued different knowledge based on the visual presence of a truth table[17]. In this study, students were given exactly the same problem twice (except for the presence of a truth table) separated by 20-30 minutes and several other domain-specific problems. Despite solving the same problem, students adopted different procedures and revealed different conceptions[17]. Without the truth table, students would conflate concepts such as implication and conjunction and omit negated complemented variables[17]. With a blank truth table present, students were able to maintain the distinction between implication and conjunction and never omitted complemented variables. These distinctions were observed regardless of presentation order[17].

This prior work critically revealed that different representations of the same information have different affordances for the problem solver. Different representations make some knowledge explicit while other knowledge is tacitly assumed. For example, in the example of truth tables, the representation affords and explicitly encourages the problem solver to exhaustively explore every case, whereas if the problem solver is simply asked to derive a Boolean logic expression, the need to exhaustively explore cases is implicit. For students with unreliable conceptual understanding, the use of exhaustive search strategies was necessary to remember the cases that distinguished implication and conjunction and to notice that complemented variables were omitted[17].

These prior studies are limited in that they focused solely on the representation that the researchers presented to the students and not on how students produced their own representations or how students' own representations constrained or enhanced their thinking. To begin studying the differences between how engineering novices and experts use and produce sketches, this following study sought to answer the following research question: How do students produce sketches when designing sequential circuits?

3. Methodology

To further explore the importance of the affordances of different representations in engineering problem solving, we are specifically examining how students and faculty differentially solve problems that explicitly require the use of multiple representation transformations. In this paper, we present initial findings from our investigations into how students and professors transform finite state machine diagrams into sequential circuits.

3.1. <u>Terminology, Concepts, and Diagrams</u>

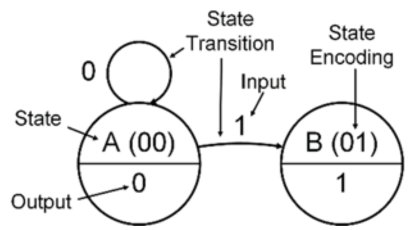

Figure 1: Partial state diagram with parts of the state machine labeled

A finite state machine (FSM) consists of a finite set of *states* S, a finite set of *input* symbols I, a finite set of *output* symbols O, a *transition* function $\delta: S \times I \rightarrow S$, and an *output* function $\omega: S \rightarrow O$ (assuming a Moore model finite state machine). The transition function maps each pair of a state and an input symbol to a *next state* in S. The output function maps each state to an output symbol. These functions can be represented graphically using a *state diagram* as shown in Figure 1 or diagram 1 in Figure 2. During the design of a FSM, the states are typically labeled with some meaningful name to facilitate interpretation of the diagram. When designing a sequential circuit, each state is given an arbitrary and unique binary state encoding. These encodings each have the same number of binary bits (i.e., fixed width) and can be selected to optimize the design of the sequential circuit. In Figure 1, the state on the left is encoded as 00. When the system is in state 00, the system will output the value of 1. An input value of 0 during this state will cause the system to transition back to state 00. An input value of 1 during this state will cause the system to transition to state 01. A diagram key is often included as part of the diagram to indicate which variable names are assigned to the state encodings, input encodings, and output encodings (far right portion of diagram 1 in Figure 2). Encodings are generally indexed in descending order so that the indices represent the power of two assigned to each position, enabling the encodings to be interpreted as unsigned binary numbers (Q_1Q_0 = 10 is read as “state 2”). The current state Q_1Q_0 is distinguished from the next state $Q_1^+Q_0^+$ with the use of a superscript ‘+’ sign.

To design a sequential circuit, the problem solver generally (though not always) needs to translate the state diagram into a tabular form, often a next-state table (see diagram 2 in Figure 2). In the next state table, state encoding variables are typically assigned to the left-most columns with input variables in the next column to the right. The next states and outputs are then listed in the following columns as they are functions of the current state and inputs. The problem solver chooses a state (e.g., state A with state encoding 00 in Figure 2) and determines the output based on the current state (e.g., output 0 for state A) and the next state based on the current state and value of the input (e.g., the top row of the next state is 01 for state A).

The tabular representation is then translated into a set of Boolean expressions (see diagram 3 in Figure 2) that determines how the state transitions and outputs are implemented. Assuming that D-type flip-flops are used in the circuit, the next-state encoding and the D input of the flip-flops are equivalent, so D is often used in place of Q^+ in these equations. Boolean expressions can be simplified to minimize the number of operators in the expression using a variety of techniques.

Finally, the Boolean expressions are used to construct a schematic for the sequential circuit. The state variables are stored in circuit devices called flip-flops (the boxes labeled FF) that has a device input (D) and device output (Q). A flip-flop stores its state until the system clock triggers it to change its state to the value being sent to its input D. This structure enables the system to be stable even though it has feedback loops. AND, NOT, and OR operators in the Boolean expression are translated into logic gates (D-shaped, triangle-shaped, and arrowhead-shaped respectively). The device-level outputs of these gates are used to compute the D input of the flip flops and the system output.

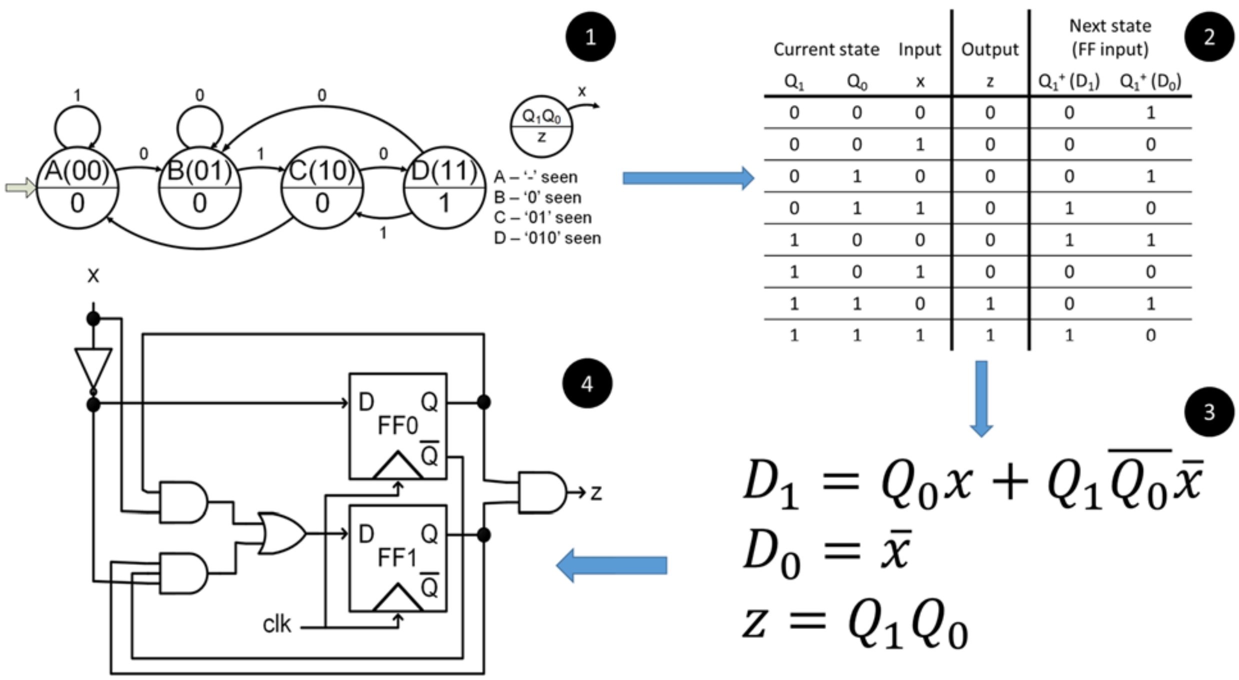

Figure 2: The series of transformations that a subject would likely use to design a sequential circuit. Sub-figure 1 is a state diagram, sub-figure 2 is a next-state table, sub-figure 3 is a set of Boolean expressions, and sub-figure 4 is a circuit diagram

3.2.<u>Sampling</u>

In Fall 2014 and Spring 2015, the authors interviewed 11 and 16 undergraduate students, respectively, at the University of Illinois at Urbana-Champaign. Additional interviews were added until the interviewers perceived that the interviews provided no new insights into students' problem solving processes. Students were recruited each semester from a large-enrollment digital logic and computer architecture course taught in the Department of Electrical and Computer Engineering. The course was lecture-based and administered weekly paper-based homework assignments and bi-weekly CAD-based design assignments. The course administered three midterm examinations and a final examination with one midterm focused specifically on the design of finite state machines. All interviewed students were traditional-age (18-22 years old) undergraduates majoring in electrical or computer engineering and had passed the digital logic course the semester before being interviewed.

All interviewed students and instructors are referred to as *subjects.*

3.3. Interview process

Subjects were interviewed for one hour. Interviews were conducted in a modified "think-aloud" format: Subjects were instructed to vocalize their thoughts as they solved problems and responded to questions. Prior to the interview, subjects were briefed on the study's goal of understanding how students and instructors design sequential circuits. They were told not to expect feedback during the interviews about whether their designs were valid, but to expect frequent requests to elaborate on what they were doing.

All interviews were conducted on a tablet computer running Windows 8 and Microsoft OneNote. The tablet provided a digitizer stylus with an eraser to create a near pencil-and-paper experience in the digital environment. Interviews were recorded using Camtasia for screen capture and audio.

Subjects were paid for their participation, and all subjects gave written consent to be interviewed under IRB approval (University of Illinois at Urbana–Champaign number 15065).

3.4. Interview questions

All subjects were interviewed using the same protocol. The interview consisted of two portions: the training tasks and the problem solving tasks.

The subjects were given a training task to familiarize them with the digital pen and tablet environment, to the process of vocalizing their thoughts, and to collect baseline data on subjects' drawing abilities and patterns. Subjects were informed that the task was a training task. They began by drawing shapes (lines, arrows, squares, etc.), the alphabet, and numbers 1 to 10. Subjects then copied a cartoon drawing of a banana (See Figure 3). The final training task asked subjects to copy a simple, computer-generated state diagram (See Figure 4).

Figure 3: Cartoon banana used for the training task

For the problem solving tasks, subjects were given four different design tasks of increasing difficulty. The first three design problems can be considered canonical design problems with known strategies for their completion. Each of these problems was intended to be analogous if not isomorphic to problems that subjects should have seen during their coursework and should be solvable in a few minutes by an expert. Notably, the sequence recognizer circuit is a conceptualized structure of a specific state machine that has such a standard final design that it can be designed by an expert immediately without a state diagram or any transformations. The fourth design problem is a complex, ambiguous design problem that while solvable is not intended to be solvable during the interview. The goal of this problem was to observe what the problem solvers did when they did not know what to do.

First, subjects were asked to design a sequential circuit based on the state diagram that they copied during the training task. This problem can be reliably solved using only the procedures described in Section 3.1. Second, subjects were asked to design a sequential circuit for a *counter* circuit (a circuit that repeatedly counts through a series of numbers) but were not given a state diagram to seed their problem solving. Third, subjects were asked to design a *sequence recognizer* circuit (a circuit that detects a targeted sequence of bits such as 0,1,1,0) but again were not given a state diagram to seed their problem solving. These latter two problems allowed us to observe how subjects constructed their own state diagrams before they engaged in the transformation process. Finally, subjects were given a food scale along with a description of the food scale's behavior. Subjects were asked to design a finite state machine that could replicate the behavior of the given food scale.

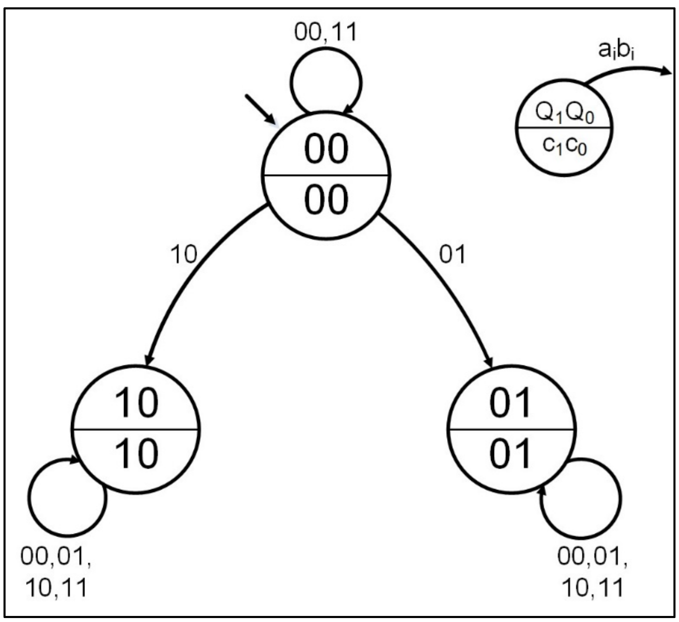

Figure 4: State diagram that students were instructed to copy during the training task

3.5. Analysis Method

Interviews were analyzed using the *constant comparative method*[22,23] without an a priori coding scheme, but with an awareness of prior research on conceptual understanding, problem solving, and comprehension of diagrams. First, two members of the research team analyzed the interviews separately to increase their familiarity with the data as well as to provide a record of their personal observations and biases. Analysis was performed directly from the video data using MaxQDA so that verbal statements could be linked to sketching behaviors. Second, the research team convened to reconcile codes. This process of reconciliation progressed sequentially through the interviews. Analysis and comparisons were made along four units of analysis of different granularities[22]: 1) the subject, 2) the problem, 3) the transformation and 4) the statement.

Subject unit of analysis: Each subject was categorized as being uninformed, a mid novice, an advanced novice, or an expert based on the accuracy and quality of their solutions across the interview. This categorization was made by consensus between the researchers.

Problem unit of analysis: Analysis of each subjects' solution was described holistically. For example, a student might reveal a robust method for designing FSMs but a poor conceptual understanding of counter circuits.

Transformation unit of analysis: Analysis focused on how a subject revealed their conceptual knowledge within a specific transformation task. For example, a subject might reveal a correct understanding of state while drawing a state diagram, but reveal an incorrect understanding of state while drawing a circuit diagram.

Statement unit of analysis: Each statement a student made or figure they drew was analyzed to document a students' conceptual understanding as revealed in the moment. For example, a subject might refer to the output of the circuit (O) as the next state (Q+) revealing a contextual conflation of the two concepts.

The goal of the constant comparative method is to make comparisons within and across these units of analysis to collect evidence for and against emergent themes and theories[22]. In this analysis approach, the researchers focus on maintaining thick descriptions of observations to facilitate comparisons rather than on reducing these observations to a strict coding scheme[22]. In the description that follows, a theme is a holistic description of an observed behavior that held true across multiple units of analysis and across multiple subjects. A theory is a framework that organized recorded themes into an interpretive narrative.

In the absence of a formal coding scheme, the authors kept an audit trail of emergent themes and theories to increase trustworthiness and reduce bias in the analysis. Throughout analysis, the authors recorded which themes and theories were generated or refined in the audit trail. Both researchers needed to agree to the inclusion of all themes and theories in the audit trail. Once a theme or theory that was admitted into the audit trail, the researchers were required to test those theories and themes with each successive unit of analysis.

4. Results

In this section, we present the first theme that emerged from the analysis of the data: *Sketching order and arrangement reveals the use of conceptual understanding.* The core assertion of this theme is that the use of conceptual knowledge during a sketch is revealed by the order in which parts of the sketch are produced as well as the physical arrangement of those parts.

4.1. Sketching behaviors of unrelated shapes, letters, and numbers vs. related parts of a banana

The training task revealed that subjects adopted different sketching behaviors when drawing a series of shapes and letters that had no particular conceptual connection than when they were drawing an object that had a conceptually linked but distinct parts (a partially peeled banana). When drawing the series of shapes, letters, and numbers, subjects each drew these objects in a natural reading order (left-to-right for domestic students and top-to-bottom for some international students. In contrast, when drawing the banana, most subjects drew the banana as a series of conceptual objects. Subjects alternatively drew each part of the peel and then would draw the fruit, or they would draw the general shape of the fruit and then add peels. These alternate drawing approaches revealed that subjects conceptualized the image as having two distinct parts: the fruit and the peel. When conceptualized in this way, copying of the image focused on drawing the concepts rather than copying lines. In contrast, when there was no particular conceptual ordering to how the shapes, letters, and numbers should be arranged, drawing proceeded in a natural reading order.

4.2. Copying of a state diagram reveals deep versus shallow conceptual processing

Similar distinctions in the use of conceptual understanding were revealed when subjects reproduced the state diagram during the training task. Subjects were told to focus on practicing two tasks: vocalizing their thought processes and becoming familiar with the experience of using a digital sketch environment. Subjects expressed varying degrees of initial comfort with the platform, so some subjects vocalized about the mechanics of drawing rather than on what they were drawing while others focused on vocalizing their thought processes.

These differences in verbal protocols aligned with different approaches to copying the state diagram that was presented during the training task. Figure 5 provides the drawing order of two subjects in parallel columns with the earliest part of the drawings shown in the top frames and each successive frame progressing forward in time. The colored arrows in this figure and all subsequent figures were added by the researchers to facilitate interpretation for the reader.

The drawings in the left column were produced by a subject whose verbal protocol focused on the experience of writing with a digital pen, saying things like “this feels comfortable.” Notice that this subject’s drawing progresses from left to right across each successive frame. Of particular note is that in the second frame, the subject draws in natural reading order rather than with a conceptual structure: the subject has drawn an arrow that does not originate from a current state, but only points to a next state. This behavior was observed only in the context of the copying task and was never replicated in any context in which a subject was creating a state machine from scratch. The creation of an origin-less arrow revealed that the subject was focused on the arrow as an arrow rather as an inscription that encoded the transition from a current state to a next state. The left-to-right drawing order confirms that the subject was focused on reproducing the shapes of the state diagram and not the meaning or conceptual structure of the diagram. Critically, the subject who produced these drawings revealed the highest level expertise of any student when completing the problem solving tasks, but was not using his conceptual understanding of state machines during this task.

In contrast, the drawings in the right column were produced by a subject whose verbal protocol focused on identifying the components of the diagram, calling circles “states” and assigning names such as “inputs” and “outputs” to the various 0s and 1s in the diagram. This subject copied the source diagram in a non-linear order, drawing the states in numeric order according to their binary encodings. She also verbally sought out the diagram key after drawing the first state to help herself understand what she was copying.

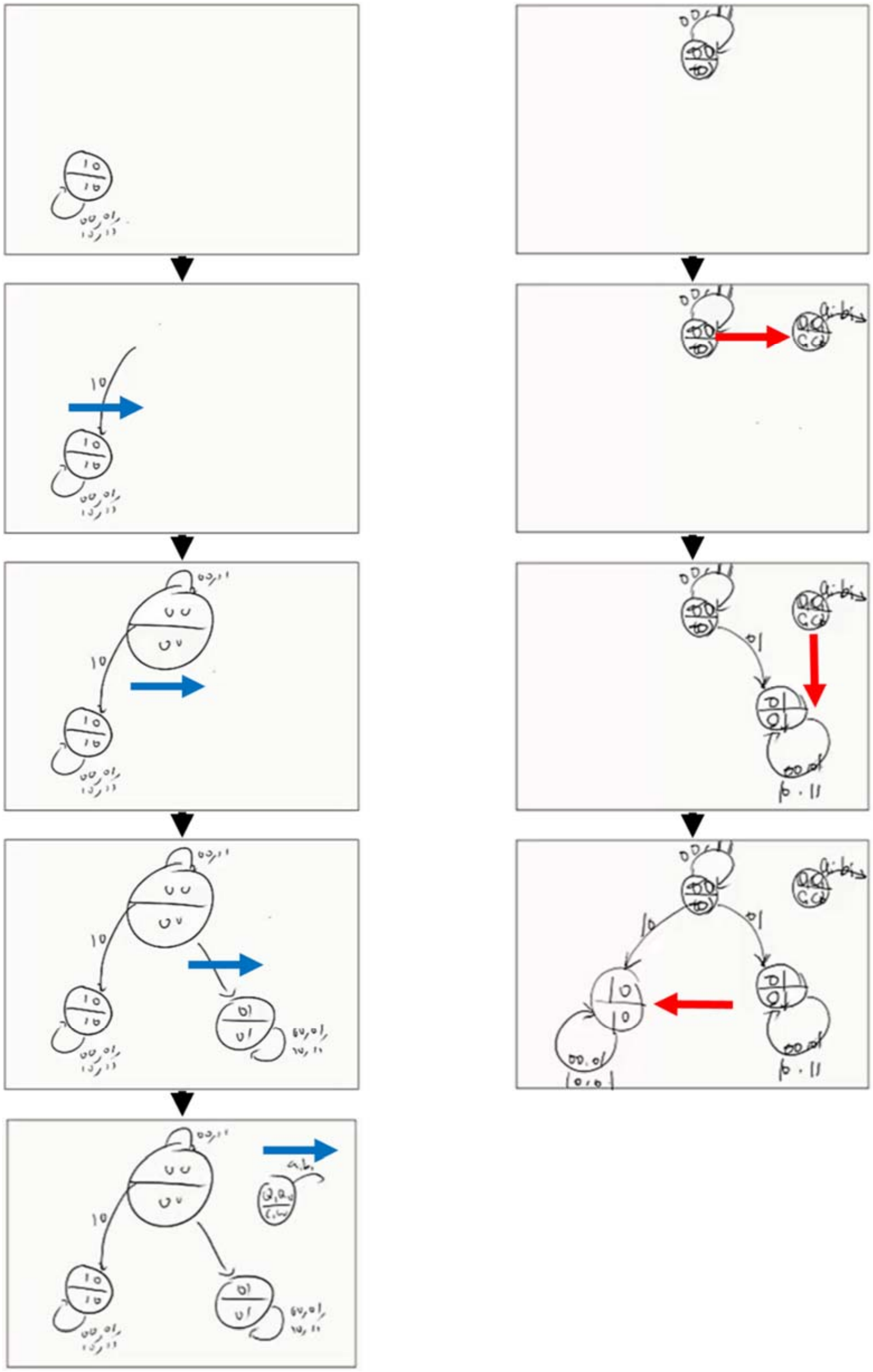

Figure 5. Left column shows the left-to-right drawing sequence of a subject focused on experiencing the drawing medium. Right column shows the non-linear drawing sequence of a subject focused on explaining the conceptual content of the diagram

4.3. Conceptual understanding differentially revealed during production of state diagrams

When creating the state diagram for the counter circuit, subjects revealed different approaches to designing the state diagram. A core concept of counter circuits is that they cyclically repeat the same sequence. The subject on the left in Figure 6 did not reveal this conceptual understanding of counter circuits during his problem solving and accordingly arranged the states of his state diagram using a left-to-right pattern. In contrast, the subject on the right arranged the states of his state diagram using a circular pattern that would support his continued use of the cyclical conception of counters.

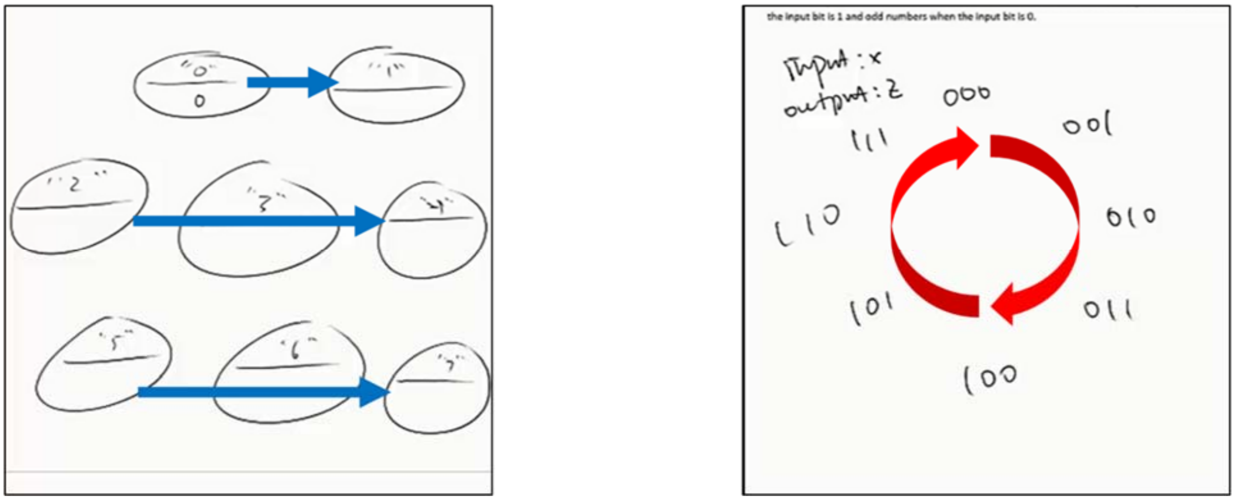

Figure 6: Subject who did not use a conception of a counter drew the states in left-to-right order (left image), while subject who did use a conception of a counter drew the states in a circular pattern.

As these two subjects continued executing their design of the state diagrams, they took different design approaches. Figure 7 shows that the subject on the left took a rigorous approach of exhaustively analyzing each state one at a time, determining the state transition for both input combinations for one state before analyzing the state transitions for a subsequent state. This approach revealed that the subject could accurately and reliably apply the problem specification to each individual state but that this subject did not approach this problem specifically as a counter problem. In contrast, the subject on the right continued to design the state diagram specifically as a counter circuit. The subject articulated that the counter behaved as an odd-number counter when the input value was 0 and drew a cycle of 0 arrows around the diagram connecting the odd-numbered states (001, 011, 101, 111 states in the top row of Figure 7). The subject then articulated that the counter behaved as an even-number counter when the input value was 1 and drew a cycle of 1 arrows inside the diagram connecting the even-numbered states (000, 010, 100, 110 states in the middle row of Figure 7). The subject concluded his analysis by drawing the remaining state transitions by switching between odd and even counting (bottom row of Figure 7).

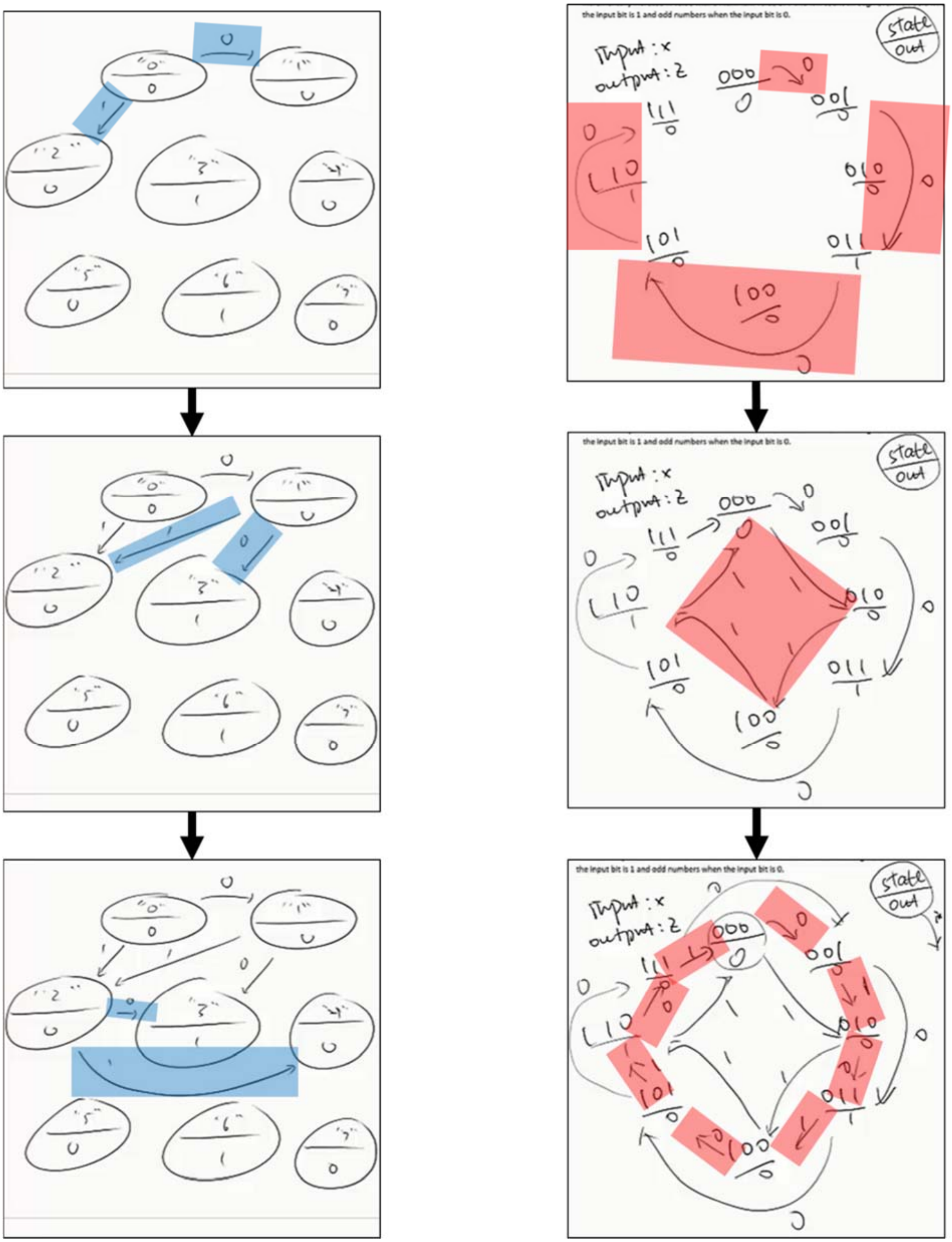

Figure 7: Left column shows how a subject who did not use a conception of a counter designed each state in numeric order. Right column shows how a subject who did use a conception of a counter designed the counter as a series of cycles.

4.4. Conceptual understanding of state revealed when drawing circuit diagrams

The use of different conceptual understanding was also revealed in the way that subjects drew their circuit diagrams. When subjects create next state tables, the state variables (Q) and system input variables (I) are treated as device-level inputs that determine the value of the next state (Q+). In the Figure 8, the top subject confuses device-level inputs and system-level inputs and treats both his state variables (S1S0) and his system inputs (I1I0) as system-level inputs. This conceptual confusion is revealed by his use of "rails" (the long vertical lines extending down from the variable names). The subject fails to maintain the conceptual focus of state as a system variable that is stored in flip flops. Without this conceptual understanding, the subjects' production of the circuit diagram progressed from left to right, ultimately forcing the student into several other conceptual mistakes such as equating the next state and the current and equating the next state and the output of the circuit.

Notably, this subject did know that state was stored in flip-flops. When asked how many flip-flops were needed to store a set of states when looking at a state diagram, the subject readily recalled the $\log_2$ relationship between the number of states and the number of flip flops needed to encode that state. These types of context-dependence of students' knowledge justify our decision to use Knowledge-in-Pieces.

In contrast the bottom subject in Figure 8, maintained the conceptual distinction between system state and system inputs and connected state to the flip flops. This subject's diagram began by drawing the flip flops, then proceeding to the left to create the next state logic for his flip flops before proceeding to the right to determine the output of the system.

5. Discussion and Conclusions

The theme *Sketching order and arrangement reveals the use of conceptual understanding* was demonstrated across each of the units of analysis. The observation was supported across subjects and within a subject's interview. The observation held across each of the problems in the interview, being revealed in the training tasks and each of the problem solving tasks. The observation also held across the different transformations, being evident in the drawing of shapes and bananas, state diagrams, and circuit diagrams. Finally, the observations were supported at the statement level as the content of the subject's verbal protocols revealed attention to different concepts or goals.

Critically, the argument we are making is that the order of sketching and the structure of the sketch are indicative of the *use* of conceptual knowledge and not whether the student *has* that conceptual knowledge. As noted in the results, the left subject in Figure 5 did not reveal conceptual knowledge about how to structure state machines during the training task, but abundantly revealed this conceptual knowledge later during the problem solving tasks. These results provide empirical evidence for the fragile or fractured structure of conceptual knowledge that causes it to be accessed differentially based on perceptual and contextual cues. Future analysis will need to further tease apart which cues are most salient in predicting the differential performance between novices and experts.

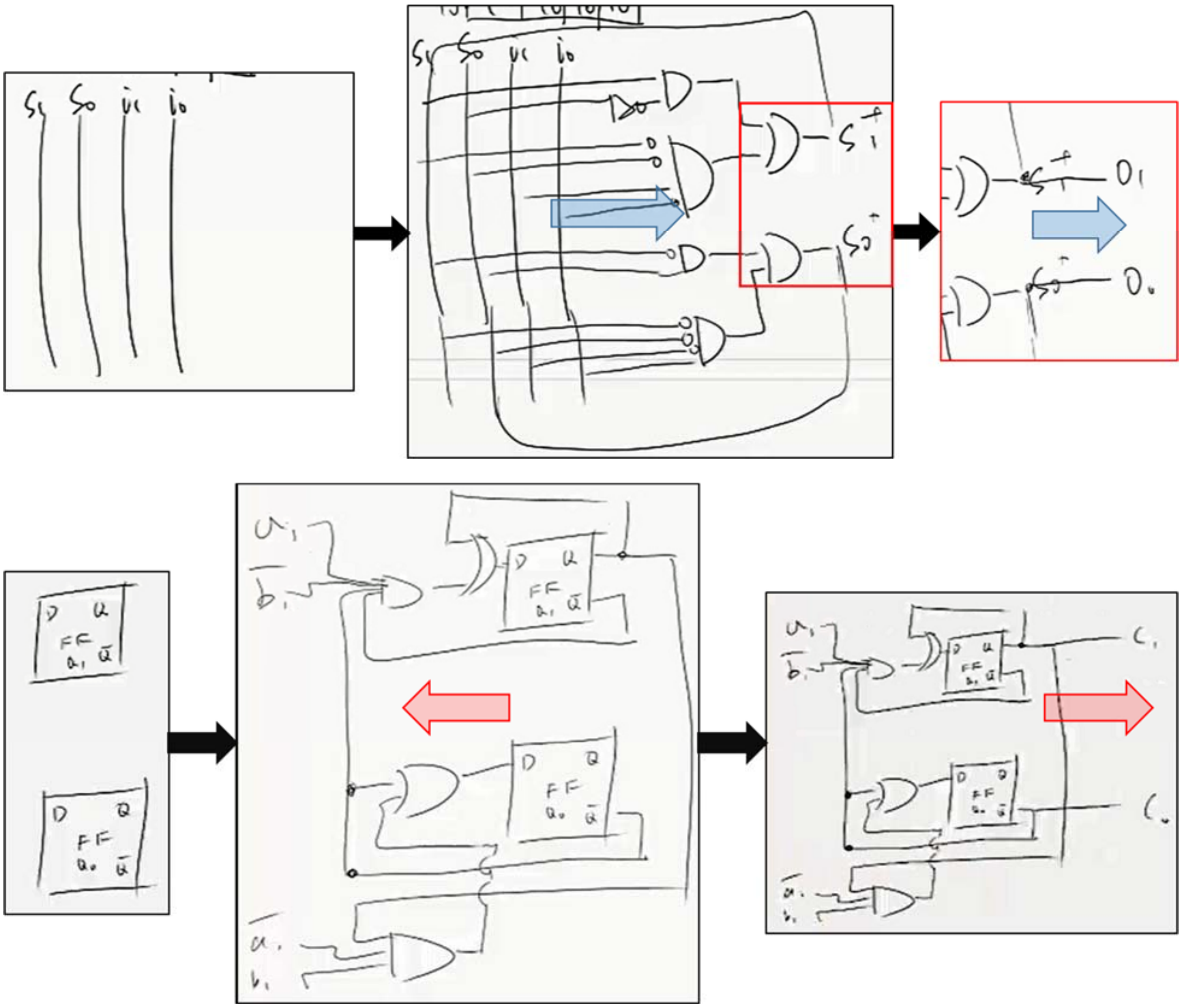

Figure 8: Top subject did not use a conception of state being stored in flip flops and progressed from left to right in his state diagram. Bottom subject used this conception and progressed to the left before progressing to the right.

The use of only left-to-right drawing and information processing may be particularly important for identifying gaps in students' knowledge or their inability to access knowledge that they possess. These gaps can suggest more targeted interventions for improving student performance. For example, the subject who used left-to-right processing of the circuit diagram failed to maintain the conceptual distinction between system-level inputs and device-level inputs. Notably during instruction, the instructors of the course rarely maintained this distinction verbally during instruction or in their course notes and assignments. Greater emphasis and the use of different vocabulary may help students maintain this conceptual distinction and improve their learning in the future.

Future work will explore whether these results are idiosyncratic to students at the University of Illinois or to the topic of state machines in digital logic. Interviews with instructors of digital logic courses are ongoing. Comparisons between the reasoning and problem solving approaches of students and instructors will be compared in future studies to enable comparisons between experts and novices.

6. Acknowledgments

Thanks to Lance Pittman for his help with collecting data and supporting analysis. This project was supported by the National Science Foundation under grant EEC 1429348. The opinions, findings, and conclusions presented in this paper do not necessarily reflect the views of the National Science Foundation or the authors' institution.

References

1 Juhl, J. & Lindegaard, H. Representations and visual synthesis in engineering design. *Journal Of Engineering Education* **102**, 2-19 (2013).

2 Jonassen, D., Strobel, J. & Lee, C. B. Everyday problem solving in engineering: Lesson for engineering educators. *Journal of Engineering Education* **95**, 139-151 (2006).

3 Litzinger, T. *et al.* A cognitive study of problem solving in statics. *Journal of Engineering Education* **99**, 337-353 (2010).

4 Henderson, K. *On line and on paper: Visual representations, visual culture, and computer graphics in design engineering.*, (MIT Press, 1999).

5 Moore, T. J., Miller, R. L., Lesh, R. A., Stohlmann, M. S. & Kim, Y. R. Modeling in engineering: The role of representational fluency in students' conceptual understanding. *Journal Of Engineering Education* **102**, 141-178 (2013).

6 Johri, A. & Lohani, V. K. Framework for Improving Engineering Representational Literacy by Using Pen-based Computing. *International Journal of Engineering Education* **5**, 958-867 (2011).

7 Tront, J. G. Facilitating pedagogical practice through a large-scale tablet PC deployment. *Computer* **40**, 62-68 (2007).

8 Johri, A., Roth, W.-M. & Olds, B. M. The role of representations in engineering practice: Taking a turn towards inscriptions. *Journal of Engineering Education* **102**, 2-19 (2013).

9 Bransford, J. D., Brown, A. L. & Cocking, R. R. *How People Learn*. (National Academy Press, 2000).

10 Slotta, J. D. & Chi, M. T. H. Helping students understand challenging science topics through ontology training. *Cognition and Instruction* **24**, 261-289 (2006).

11 Yang, D. *et al.* Repairing student misconceptions using ontology training: A study with junior and senior undergraduate engineering students. in *Proceedings of the 2010 American Society for Engineering Education Annual Conference and Exposition* (2010).

12 Hestenes, D., Wells, M. & Swackhamer, G. Force concept inventory. *The Physics Teacher* **30**, 141-158 (1992).

13 Clement, J. Using bridging analogies and anchoring intuitions to deal with students' preconceptions in physics. *Journal of Research in Science Teaching* **30**, 1241-1257 (1993).

14 Dufresne, R., Gerace, W. J., Hardiman, P. T. & Mestre, J. P. Constraining novices to perform expert-like problem analyses: Effects on schema acquisition. *Journal of the Learning Sciences* **2**, 307-331 (1992).

15 Chi, M. T. H., Feltovich, P. J. & Glaser, R. Categorization and representation of physics problem by experts and novices. *Cognitive Science* **5**, 121-152 (1981).
16 Jee, B. D., Gentner, D., Forbus, K., Sageman, B. & Uttal, D. H. Drawing on experience: Use of sketching to evaluate knowledge of spatial scientific concepts, in *Proceedings of the 31st Annual Conference of the Cognitive Science Society* 2499-2504 (2009).
17 Herman, G. L., Loui, M. C., Kaczmarczyk, L. & Zilles, C. Discovering the what and why of students' difficulties in Boolean logic. *ACM Transactions on Computing Education* **12**, 3:1-28 (2012).
18 Herman, G. L., Zilles, C. & Loui, M. C. Flip-flops in students' conceptions of state. *IEEE Transactions on Education* **55**, 88-98 (2012).
19 Montfort, D. B. *et al.* Patterns of student conceptual understanding across engineering content areas. *International Journal of Engineering Education* **31**, 1587-1604 (2015).
20 Montfort, D., Herman, G. L., Streveler, R. & Brown, S. Assessing the application of three theories of conceptual change to interdisciplinary data sets. in *Proceedings of the Forty-Second ASEE/IEEE Frontiers in Education Conference* (Seattle, WA, 2012).
21 Herman, G. L., Montfort, D., Brown, S. & Streveler, R. Work-in-progress: Do students need to learn to speak engineering-ese?? Conceptual change as language acquisition in engineering. in *Proceedings of the Forty-Second ASEE/IEEE Frontiers in Education Conference* (Seattle, WA, 2012).
22 Boeije, H. A purposeful approach to the constant comparative method in the analysis of qualitative interviews. *Quality & Quantity* **36**, 391-409 (2002).
23 Corbin, J. & Strauss, A. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. (Sage, 2007).