Augmented Proof: Examining Structures to Support Geometric Proof Comprehension

Anonymous CogSci submission

Abstract

To understand even a modest geometric proof, students must process an interwoven combination of symbolic, diagrammatic, geometric, and logical information. This amount of information density presents a daunting management task that students are known to perform poorly on. To address this challenge, we propose a two-column proof interface that structures the information management task according to *diagram configuration schemas* (DCS) (K. R. Koedinger & Anderson, 1990a). We evaluated our design by comparing secondary school students' performance on proof comprehension tasks with DCS augmentation to using a typical two-column proof. Students using the DCS-augmented interface demonstrated improved overall reasoning and accuracy in geometric proof tasks compared to the traditional two-column format. They were also significantly better at identifying and correcting mistakes in proofs. These results suggest that managing complex information by integrating it in a coherent schema—like DCS—can support student understanding of proof.

Keywords: geometry; proof visualization; interactive interfaces; perceptual chunking; diagram configuration schemas.

Introduction

In U.S. math curricula, secondary school geometry is a student's first introduction to proof which is widely regarded as a challenging topic for students (Senk, 1985; Stylianou et al., 2009; Usiskin, 1982). Despite many advances in the geometry curriculum, the artifacts that students use to learn proof have remained largely unchanged for several decades. Most students still do proofs using the traditional two-column format, which presents steps in a linear progression and leaves key information like dependencies between steps implicit (Soucy McCrone & Martin, 2004). However, the logical flow of proof is not linear, as some steps do not need justification and others have multiple prerequisites. Mentally tracking all of this information on top of recalling relevant geometric rules, terminology, and symbols is a complex management task that many students never master (Mayberry, 1983).

Diagram configuration schemas (DCS) are established memory structures that associate a diagram with valid inferences and conditions for those inferences (K. R. Koedinger & Anderson, 1990a). DCS allows experts to visually pattern-match, simplifying the search space and solving proofs using fewer inferences than novices. Therefore, students—who should learn to build these memory structures for themselves—could benefit from DCS-informed cues as they practice.

Figure 2 shows the Congruent-Triangles-Shared-Side schema from the original publication (K. R. Koedinger & Anderson, 1990a). DCS describes recognizable patterns, where finding certain *configurations* with enough *part-statements* means that that all other statements in the schema *can* be proven. Take the proof of Figure 1 as an example. Note that Figure 2 accurately describes the configuration of the proof shown in Figure 1. In fact, steps 1 and 2 perfectly match with part-statements 1 and 2 in Figure 2, and thus fulfill the first "way-to-prove." An expert with knowledge of DCS is convinced that the proof can be solved within the first two steps. Our work differs from DCS in that the task is to comprehend proofs that are already filled out. This requires a level of granularity that DCS does not capture. In a sense, DCS treats common minutia such as using the Reflexive Property or the differences between SAS and SSS as assumptions that are correctly applied outside the scope of the schemas, but they are critical details in the classroom setting. A complete and correct two-column proof at the secondary school level *should include* the additional steps that experts skip over.

To fit the benefits of DCS within the context of these more granular proofs, we adapt the components of DCS to represent information at the level of geometric reasons: theorems or definitions that students are familiar with and use in class. For each reason, we show the configuration, the whole-statement, the part-statements, and compare valid ways-to-prove with the part-statements that are established by the proof. This allows us to structure attention around the task of evaluating whether there are enough ways-to-prove the whole-statement at each step. This adaption of DCS defines the information to be highlighted in order to visualize the state of each step. This is what we call ***DCS augmentation (DCSA)***.

The addition of DCSA may structure student focus and help them perform better on proof comprehension tasks. It is also possible that adding visual cues to an already complex processing task could add extraneous cognitive load (Sweller, 1994, 2010). This could result in students becoming more confused or overwhelmed and hurt their performance. These hypotheses motivate our research question: ***Does adding DCSA to two-column proofs result in improved accuracy on geometric proof comprehension tasks at the secondary school level?*** To answer this question, we build interfaces for both DCSA and traditional two-column proofs (serv-

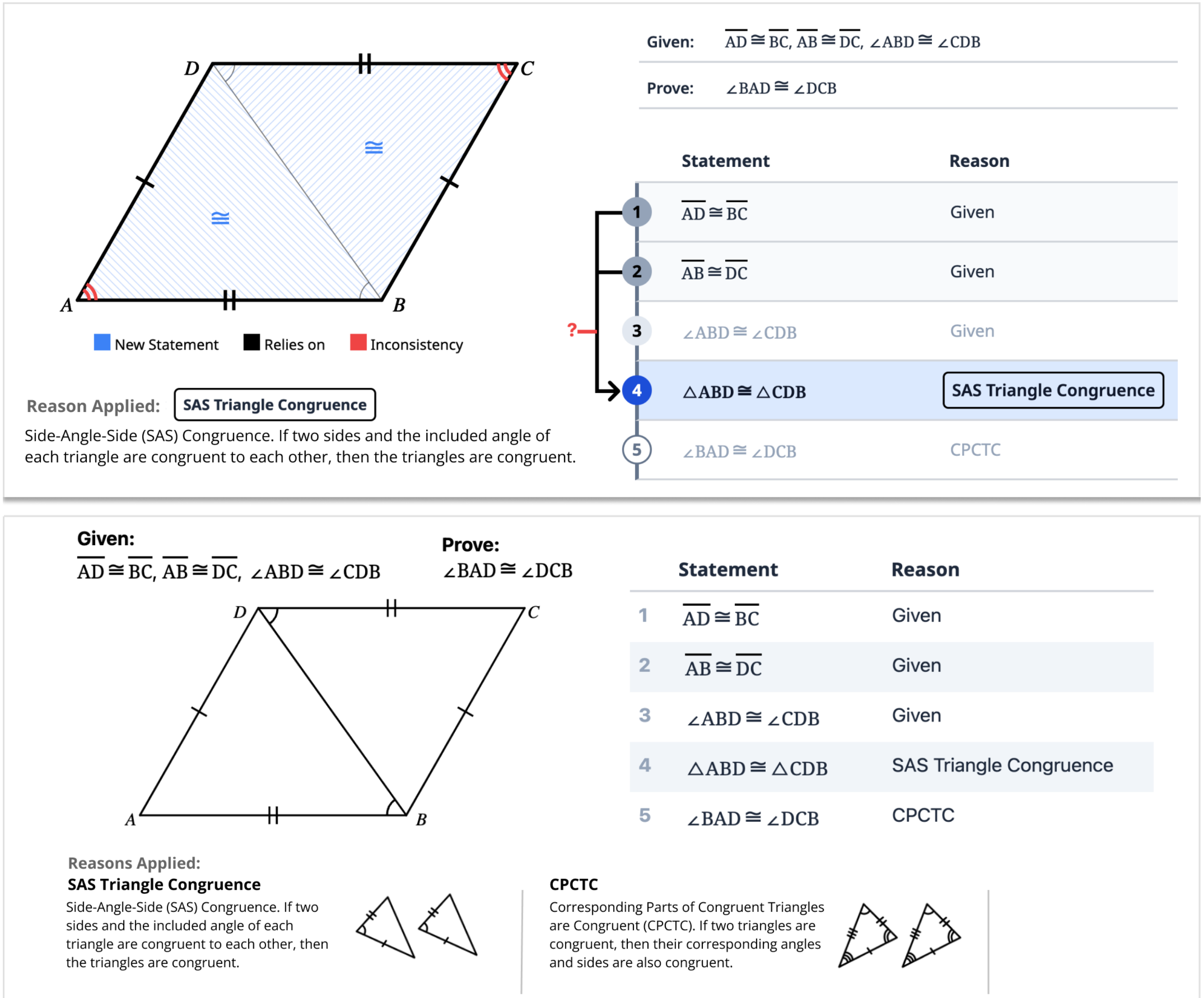

Figure 1: **Top**: Screenshot of the DCS augmented interface displaying the fourth step of a proof. Students use the up and down arrow keys to navigate through the proof and see changes in state, definitions, and dependencies. This proof mistakenly applies the Side-Angle-Side (SAS) Triangle Congruence theorem without enough prerequisite information. The red bolding indicates that the interior angles necessary for SAS are not known to be congruent by step 4. Adding a reflexive step $\overline{BD} \cong \overline{BD}$ would correct this proof. **Bottom**: Screenshot of the control interface of the same incorrect proof, providing all the information necessary to comprehend the proof simultaneously. It is akin to the proof format that secondary school students practice.

ing as a control condition representing standard classroom practice) and run a between-subjects, think-aloud experiment with 13 U.S. secondary school participants. Our study compares the performance of students on various proof comprehension tasks with these interfaces. While both interfaces present equivalent *geometric* information about the same set of proofs (Larkin & Simon, 1987), the DCSA interface includes a layer of visual cues to highlight components of each applied DCS.

In the following sections, we discuss the design of the DCSA interface including how DCS-specific information is implemented. Then, we discuss the methodology, including our selection of proofs and comprehension tasks. We present quantitative and qualitative results from the think-aloud experiment and close with a discussion on the findings, implications, and limitations of our study.

Interface Design

Components of a two-column proof The typical components of a two-column proof are illustrated in Figure 1-Bottom. The premises include three pieces: given information, the goal to prove, and the **construction**, which illustrates the objects in the proof. The two-column format lists steps comprising **statements** and **reasons**. A reason must justify each statement, and each reason must be either given or a known geometric rule (i.e., theorem, definition). In the exam-

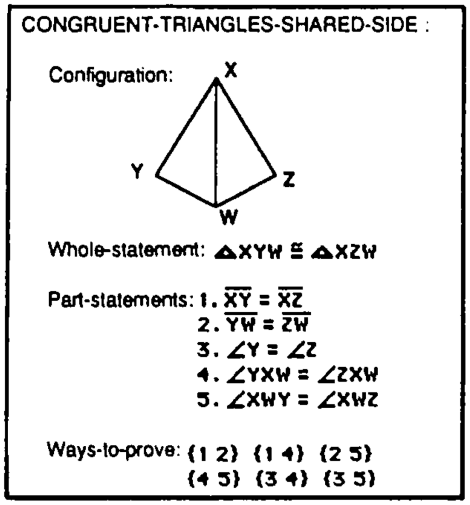

Figure 2: The diagram configuration schema that corresponds to the proof in Figure 1. The numbers in the ways-to-prove indicate part-statements. $\{1\,2\}$ means that if the part-statements $\overline{XY} = \overline{XZ}$ and $\overline{YW} = \overline{ZW}$ are proven, all the statements of the schema can be proven (K. R. Koedinger & Anderson, 1990b).

ple here, the proof incorrectly proves that the two triangles are congruent by "Side-Angle-Side Triangle Congruence" (SAS) to leverage the "corresponding parts of corresponding triangles are congruent" rule (CPCTC). SAS needs two pairs of congruent sides and an *included* pair of congruent angles. Instead, the givens depict Side-Side-Angle, which is inadequate to prove triangle congruence. Adding the statement $\overline{BD} \cong \overline{BD}$ by the Reflexive Property before step 4 would correct this proof.

This two-column interface serves as the control for the experiment. It has the standard geometric information of a two-column proof and also contains a list of illustrated definitions of all the geometric reasons used within the proof. These are intentionally drawn in a different context than the proof. This is to match the standard experience of students today, where looking up definitions online or in a textbook provides text explanations and illustrations in different contexts than the proof they are solving.

DCS Augmentation The DCS augmented interface Figure 1-Top contains the same geometric proof information as the control interface. The key addition is DCSA—visual cues designed to highlight key components of the proof at each step. Navigating with the up and down arrow keys shows DCSA information for the current step, which includes cues based on the components in a DCS: whole-statement, part-statements, configuration, and ways-to-prove. For each of these components, we refer to Figure 1-Top to describe where they are located on a two-column proof and how they are highlighted with DCSA.

Highlighting whole-statements For a step in a two-column proof, the whole-statement is its statement (i.e., $\triangle ABD \cong \triangle CDB$ for step 4). Valid whole-statements are highlighted on the construction with **blue bolding**. When the proof makes an incorrect whole-statement with the assigned configuration schema (such as claiming that all angles in triangles are equal when it is not known if the triangles are equilateral), the whole-statement is highlighted with **red bolding** on the construction.

Highlighting part-statements For a step in a two-column proof, part-statements are any previously established statements (i.e., statements of steps 1, 2, and 3 are all part-statements for step 4). Conclusions about segments, angles, and triangles can be visually indicated with tick marks and are often needed for later steps in a proof. Therefore, whole-statements become part-statements for subsequent steps and are visually indicated by maintaining the tick marks for the rest of the proof. This behavior is consistent with how students are taught to annotate their diagrams while they work through proof.

Highlighting ways-to-prove In a two-column proof, the ways-to-prove are the prerequisites for a reason to be applied. These prerequisites are normally left unstated. In DCSA, part-statements that contribute to ways-to-prove used in the proof are indicated by connecting them to the current step with a black arrow, and by being **bolded in black** on the construction (i.e., the statements of step 1 and 2 contribute to proving SAS). If a requirement of a way-to-prove is missing from the proof, a red "?" is shown, and the missing part-statements are **bolded in red** on the construction (i.e., the only angles that would complete the chosen way-to-prove are the red $\angle A$ and $\angle C$ which have not been established as congruent by step 4 of the proof). The number of arrows drawn for a way-to-prove matches the number of requirements of the reason (i.e., SAS has 3 requirements, thus there are 3 connections). These arrows are similar to the connections in a flow chart proof (Miyazaki et al., 2014).

The configuration The configuration is the top-level encapsulation of the schema, combining the knowledge about what is being proven, and what previous (or missing) information contributes. For a step in a two-column proof, the configuration is its reason (i.e., SAS Triangle Congruence for step 4). It is visually represented as the combination of the whole-statement and ways-to-prove cues (i.e., SAS is applied to prove $\triangle ABD \cong \triangle CDB$, indicated with blue fill, while the requirements for SAS are visually indicated by the black bolding of the congruent segments $\overline{AD} \cong \overline{BC}$, $\overline{AD} \cong \overline{BC}$, and the missing part-statement $\angle A \cong \angle C$). Note that this combined representation of the configuration is equivalent to drawing the illustrated reasons of the control interface (Figure 1-Bottom) over the construction and highlighting differences with red.

Red cues on either the construction or the proof indicate to the student that an issue requires investigating. To fully understand *why* something is red, students must develop their own geometric reasoning of the mistake, as the error could be in either the whole-statement or the way-to-prove. As shown in the results, we qualitatively analyze the think-aloud data to ensure that students completing tasks comprehend the proofs geometrically and do not answer questions based on visual cues alone.

Method

Design and Procedure We evaluated the effectiveness of DCSA with a between-subjects experiment. The independent variable is the assigned interface—either DCS augmented (Figure 1-Top), or control (Figure 1-Bottom)—and our dependent measure is task accuracy. Each student participant was randomly assigned to one of these two conditions. Participants were instructed to complete the tasks to the best of their ability while thinking aloud (Chi et al., 1994). The procedure followed the sequence: (1) A 10-minute pretest established their prerequisite knowledge with fundamental concepts needed for reading and understanding geometric proof. This knowledge included understanding the concept of congruency and recognizing visually congruent segments, angles, and triangles. (2) An ungraded, 5-minute tutorial facilitated by a member of the research team where they were walked through two examples of their assigned interface, introduced to the features, and given time to familiarize themselves with thinking aloud. (3) The main think-aloud procedure for eight geometric proofs in their assigned interface. This procedure had a 35-minute time limit with a 7-minute cap on each proof. (4) At most a 10-minute semi-structured interview about their experience where they were asked to reflect on any difficulties they encountered.

We created 8 geometric proofs involving triangle congruence, plus 2 additional short proofs for the tutorial. Participants viewed a correct proof and an incorrect proof in the tutorial, and 3 correct proofs and 5 incorrect proofs in the procedure. The errors were intentionally diverse: two proofs used the wrong geometric reason to skip a step later in the proof; one used the wrong geometric reason when another would have sufficed; 3 claimed objects were congruent without appropriate justification to use them later in the proof. The constructions associated with the 10 proofs were illustrated symmetrically to catch students who make assertions based on what visually appears to be true (Soucy McCrone & Martin, 2004). For example, in Figure 1 $\triangle ABD$ and $\triangle CDB$ appear equilateral, tempting students to conclude that all angles and sides are equivalent without proof. All participants viewed a random ordering of the same eight (plus two tutorial) proofs and only saw their assigned interface.

Some existing assessments of proof comprehension include proof-checking tasks, such as (K. Koedinger, 1991; Soucy McCrone & Martin, 2004). Based on the tasks in these assessments, we designed a set of 38 questions to evaluate proof comprehension and refined them over multiple rounds of pilots. The goal of each task was to exercise the student's understanding of proof by asking both surface-level questions and ones that require a comprehensive understanding of the proof. Students were asked to evaluate the correctness of every proof (question type, or QT1). If they believed the proof had a mistake they were asked to specifically identify it (QT2) and correct it (QT3). Afterward, they were asked 1-2 proof-specific questions, either to consider the validity of an alternate proof order (QT4) or to assess the final state of the proof (QT5). Every participant answered at least 22 questions, plus an additional 2 for each proof they thought was incorrect, for a maximum of 38 questions. By having students complete these tasks while thinking aloud we could determine how students approach new proofs and pinpoint their misconceptions (Chi et al., 1994).

Table 1: Participant demographics for each condition. Age, Grade, and Years Since Geometry were recorded at the time of their participation. A "0" in Years Since Geometry indicates that the student was currently taking their geometry class.

	Participant	Age	Grade	Years Since Geometry
DCSA	SP1	15	10	0
	SP3	15	10	1
	SP5	15	10	1
	SP7	15	10	0
	SP9	15	10	0
	SP11	16	11	2
	SP13	14	9	0
Control	CP2	16	10	1
	CP4	17	12	3
	CP6	15	10	1
	CP8	17	12	3
	CP10	15	10	0
	CP12	15	9	0

Participants 13 U.S. secondary school students who had learned triangle congruence theorems and geometric proof were paid to participate in the experiment. P7 was removed from the final results for not meeting the screening criteria for participation. Specifically, they did not understand the concept of triangle congruence and said they were randomly guessing through most of the procedure. The mean pretest scores for participants in each group was 81% (SD=11% for participants using DCSA, SD=7% for participants using the control), indicating that both groups had similar levels of prerequisite knowledge.

Apparatus and Stimuli The experiment was conducted on a Mac laptop with an external 1440p monitor, keyboard, and mouse. The stimuli was a website built with React. Metadata about each proof including text, construction, and definitions was used by both the DCSA and control interfaces. Proofs were displayed in a sequential, random order while questions appeared in a toolbar at the top of the page. Buttons and form elements were provided within the question toolbar. Answers were recorded upon submission. Experimental sessions were captured with audio and screen recordings.

For each proof, the students were asked to try to understand it while thinking aloud before answering the questions. They were allowed to freely interact with the proof interface as they answered the questions. If they were unsure of an answer, they were instructed to make an educated guess. In addition, all participants received scratch paper to use as they wished.

Scoring Scoring was identical between conditions: 1 point was given for each question that students answered correctly, and for QT3, which involved describing how they would correct a proof, 1 point was given if the student's description corrected the proof without resulting in unnecessary steps, or if they correctly identified that the proof could not be corrected. Students who correctly answered that there was no mistake in a proof were awarded 3 points. This was done to match the possible 3-point score of identifying and fixing an incorrect proof. The total accuracy was calculated by adding the points awarded and dividing by 38 (the highest possible score).

Results

Our research question examines the effect of DCSA on proof comprehension. We ran a linear model to determine the effect of version (DCSA vs control interface) and pretest scores on overall accuracy. Version had a significant effect on overall accuracy (Estimate=.267, $p < 0.01$). Pretest scores were not significant (Estimate=0.01, $p = .97$). Students referring to DCSA had a mean accuracy of 78.3% (SD=.12), while those referring to the control only achieved a mean accuracy of 47.2% (SD=.13). The distribution of accuracies for each interface is visualized in Figure 3.

We also analyzed performance across the five question types. We used a linear mixed-effects model to analyze the effects of version (DCSA vs control interface), question type, and their interaction on participant scores. We controlled for pretest scores as a covariate and accounted for random intercepts by participant. Again, pretest scores were not a significant predictor of final performance ($F(1,9) = 0.014$,

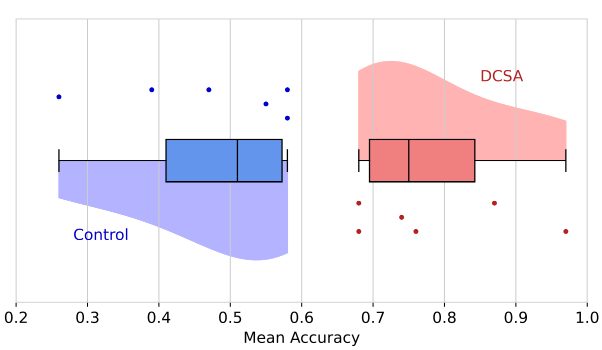

Figure 3: Raincloud plot showing distributions and per-participant accuracies for each interface. Participants using DCSA achieved a higher mean accuracy (78%) on the test questions than the participants in the control group (47%).

$p = 0.91$). We found significant main effects of version ($F(1,9) = 16.64$, $p < 0.005$) and question type ($F(4,40) = 12.40$, $p < 0.0001$), indicating that performance varied across question types. A significant interaction between version and question type was observed ($F(4,40) = 4.36$, $p < 0.005$), indicating that the performance advantage of DCSA relative to the control interface varied by question type. Specifically, participants using DCSA performed significantly better on QT1 (Estimate = 0.36, $p < 0.005$), QT2 (Estimate = 0.36, $p < 0.005$), and QT3 (Estimate = 0.30, $p < 0.01$). QT1, and QT2 are related to identifying mistakes, and QT3 involves correcting them. The raincloud plot in Figure 4 shows the differences in mean accuracy for each of the five question types.

Analysis of Geometric Reasoning To determine whether students got answers correct based on sound geometric reasoning, we first qualitatively coded all of the think-aloud transcripts and screen recording data. Every time a student answered a question, we noted whether the student provided verbal justification with correct geometric logic. Correct geometric logic was classified by the student using geometric terms to correctly describe why a statement is true or false. For instance, with the proof in Figure 1, an example of valid geometric reasoning for the question, "Is there a mistake in this proof?" would be, "Yes, because SAS Triangle Congruence is not the right reason to use at step 4." 1 point was given for each demonstration of correct reasoning out of a maximum of 38. Sometimes students provided what could be valid justification if a proof illustrated a special case (i.e., in Figure 1, a claim that "all of the angles are equal" is valid if and only if the triangles are known to be equilateral). If the special case was not established within the scope of the proof then it was marked as incorrect reasoning. In the DCSA group, students who verbally acknowledged a DCS cue were given 1 point only if they correctly explained the cue's meaning in the context of the proof, using geometric terms (i.e., in Figure 1, pointing to the **?** and then explaining that there isn't enough information to use SAS would qualify as correct reasoning). Unknown reasoning, classified as answers the student submitted without verbalization before or after submission, received 0 points. The student's reasoning score was a count of every demonstration of correct reasoning divided by 38.

Our analysis revealed that students in both conditions answered questions with some form of geometric reasoning for 90% of all tasks. The other 10% (factored into the correlation analysis as incorrect reasoning) was made up of unknown reasoning or random guesses. We used a Pearson correlation analysis to evaluate the relationship between each student's accuracy and their demonstrated reasoning. The results revealed a strong positive correlation between the two variables, $r = 0.954$, $t(10) = 10.113$, $p < 1.4 \times 10^{-6}$, indicating a significant relationship between geometric reasoning and accuracy on questions. This correlation shows that students using DCSA outperformed control interface users in both accu-

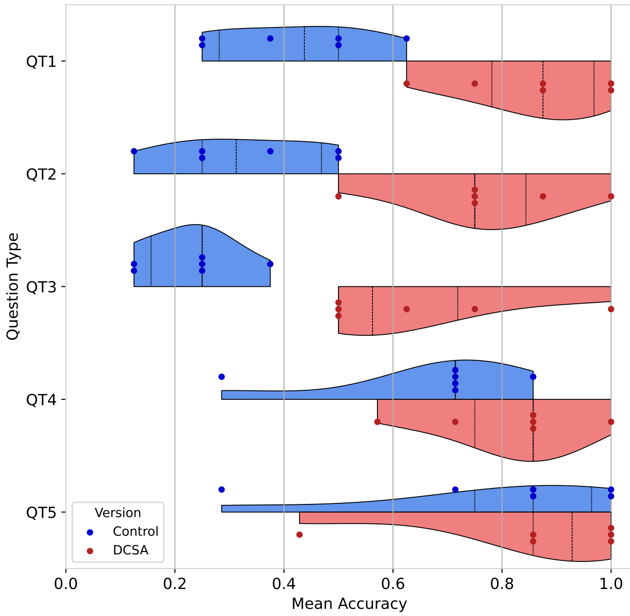

Figure 4: Raincloud plots visualizing the distribution and per-participant accuracy for each of the 5 question types. On average, participants in the DCSA group outperformed the control group in QT1 through QT4. They were significantly more accurate on QT1 through QT3 which involved finding and fixing mistakes in proofs.

racy and geometric reasoning. Furthermore, it demonstrates that students using DCSA did not simply "read off" the visual cues to obtain higher scores.

Discussion

This paper implements and evaluates a navigable format of geometric proof that includes visual cues based on the theory of DCS which we call DCS augmentation or DCSA (K. R. Koedinger & Anderson, 1990a). We use this design probe to understand whether student performance on proof comprehension tasks is significantly impacted by the addition of DCSA. Our experiment shows that DCSA and question type (both independently and together) statistically significantly impact proof comprehension accuracy. Students using DCSA did not become cognitively overwhelmed by the added visual cues; they were significantly more accurate. They also performed significantly better on tasks that involved identifying and correcting mistakes on proofs.

We qualitatively analyzed the think-aloud transcripts to address concerns that the students using DCSA only performed better because they had more information. We found a strong positive correlation between task accuracy and correct geometric reasoning, indicating they correctly engaged with the questions. Overall, our findings provide some evidence for the potential of DCSA to support student comprehension of geometric proof.

These results are consistent with previous findings that the two-column proof provides insufficient support for secondary school-level students (Soucy McCrone & Martin, 2004). We hope that our implementation can be taken as one example from a spectrum of possible levels of intermediate support between low support (i.e., the widely used fill-in-the-blank two-column proof (Soucy McCrone & Martin, 2004)) and powerful tools like the Geometry Tutor built on the ACT-R framework (Anderson et al., 1985) or the ANGLE tutor that teaches students to build flowchart representations of proofs using DCS (K. R. Koedinger & Anderson, 1990b).

Separate from our research question, we also observed some interesting qualitative trends regarding the differences in behavior between the two interfaces. The most common class of errors for both groups involved making inferences based on incorrect mental models of the proof. We classified these errors as ones where students forgot which segments, angles, or triangles were proven congruent and based their answers on an incorrect recollection. This error was more frequent in the control group, who made this mistake 42 times (60% of all errors the group made). In contrast, students in the DCSA group only had this issue 14 times (35% of all errors the group made). This trend suggests a relationship between DCSA and the cognitive processes involved in proof comprehension and warrants further investigation.

There are several limitations of our experiment. Because of the complexity of the interfaces and tasks, it was not possible to precisely determine what visual attributes particularly influenced a student's answer, only that they were able to arrive at the correct or incorrect reasoning. Future work should more accurately detail the cognitive processes that occur when referring to DCSA for direct evidence that the visual cues we applied draw attention to the components of DCS. Such a study should look for evidence that students using DCSA consider the configuration, whole-statement, part-statements, and ways-to-prove, at the same time or in close succession. This could be accomplished by obscuring parts of the interface by default and requiring students to deliberately hover or click over attributes of DCSA as they work. Though our proof comprehension test went through several rounds of pilot testing before running the experiment, the materials have not been validated by other geometry experts. Lastly, though we achieved statistically significant results, the sample size of our study is small, so we would caution against making broad claims about the general population based on this study.

Future research directions could examine the effect of DCSA on students who are tasked with completing proofs, as proof completion is the prototypical measure of a student's mastery of proof. Additionally, the underlying metadata representation of the proof that we developed for this experiment could be leveraged to provide more personalized, corrective, or explanatory feedback to students as they work.

References

Anderson, J. R., Boyle, C. F., & Yost, G. (1985). The geometry tutor. *IJCAI*, 1–7.

Chi, M. T., De Leeuw, N., Chiu, M.-H., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive science*, *18*(3), 439–477.

Koedinger, K. (1991). *Tutoring concepts, percepts, and rules in geometry problem solving* [Doctoral dissertation].

Koedinger, K. R., & Anderson, J. R. (1990a). Abstract planning and perceptual chunks: Elements of expertise in geometry. *Cognitive Science*, *14*(4), 511–550.

Koedinger, K. R., & Anderson, J. R. (1990b). Theoretical and empirical motivations for the design of angle: A new geometry learning environment. *Working notes of the 1990 AAAI Spring Symposia on Knowledge-Based Environments for Learning and Teaching, Stanford University, March*, 27–29.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, *11*(1), 65–100.

Mayberry, J. (1983). The van hiele levels of geometric thought in undergraduate preservice teachers. *Journal for research in mathematics education*, *14*(1), 58–69.

Miyazaki, M., Fujita, T., & Jones, K. (2014). Functions of open flow-chart proving in introductory lessons of formal proving. *North American Chapter of the International Group for the Psychology of Mathematics Education*.

Senk, S. L. (1985). How Well Do Students Write Geometry Proofs? [Section: Mathematics Teacher]. https://doi.org/10.5951/MT.78.6.0448

Soucy McCrone, S. M., & Martin, T. S. (2004). Assessing high school students' understanding of geometric proof. *Canadian Journal of Science, Mathematics and Technology Education*, *4*(2), 223–242. https://doi.org/10.1080/14926150409556607

Stylianou, D. A., Blanton, M. L., & Knuth, E. J. (2009). *Teaching and learning proof across the grades*. Routledge New York.

Sweller, J. (1994). Cognitive load theory, learning difficulty, and instructional design. *Learning and instruction*, *4*(4), 295–312.

Sweller, J. (2010). Element interactivity and intrinsic, extraneous, and germane cognitive load. *Educational psychology review*, *22*, 123–138.

Usiskin, Z. (1982). Van Hiele Levels and Achievement in Secondary School Geometry. CDASSG Project. Retrieved September 12, 2024, from https://eric.ed.gov/?id=ed220288