CrossMark

# Designing a Web-based Learning Support System for Flow-chart Proving in School Geometry

**Mikio Miyazaki**[1] · **Taro Fujita**[2] · **Keith Jones**[3] ·
**Yasuo Iwanaga**[1]

**Abstract**  As international research confirms, many secondary school students can find it difficult to construct mathematical proofs. In this article, we explain the pedagogical and technological underpinnings of a web-based learning support system for students who are just starting to tackle deductive proving in geometry. We show how the system was designed to enable students to access the study of proofs in geometry by tackling proof problems where they can 'drag' sides, angles and triangles from the figural diagram of the problem to on-screen cells within a flow-chart proof format. When doing so, the system automatically converts the figural elements to their symbolic form and identifies any of four kinds of errors in the learners' proof attempts, providing relevant feedback on-screen. We use empirical examples from our pilot studies to illustrate how this combination of technological features and systematic feedback can support student understanding of the structure of proof and how to plan one. Finally, we point out some limitations to mathematical expression and the usage of the flow-chart format, and indicate the prospect of minimizing such limitations by adopting a learning progression for the introductory lessons concerning deductive proofs.

✉  Mikio Miyazaki
    mmiyaza@shinshu-u.ac.jp

    Taro Fujita
    t.fujita@exeter.ac.uk

    Keith Jones
    d.k.jones@soton.ac.uk

    Yasuo Iwanaga
    yasuolucia@wit.ocn.ne.jp

1   Institute of Education, Shinshu University, Roku-Ro Nishinagano, Nagano-Shi 380-8544, Japan

2   Graduate School of Education, University of Exeter, Exeter, UK

3   School of Education, University of Southampton, Southampton, UK

🙋 Springer

In technology-based learning environments for mathematics education, the interaction between a learner and technology is based on "a symbolic interpretation and computation of the learner input [… with feedback provided] in the proper register allowing its reading as a mathematical phenomenon" (Balacheff and Kaput 1996, pp. 469–70). Much technology-related research in mathematics education continues to be concerned with the use (or absence) of all these various forms of technology in teaching, including task design and the professional development of teachers. There is a wide range of different types of technological support for teaching and learning, which, as Brown et al. (2005), p. 28) illustrate, can be categorized into mathematics teaching software (such as dynamic geometry, graph-plotting or symbolic algebra software), suitable programming languages (such as Logo), 'small programs' that address very specific aspects of the mathematics curriculum (often in the form of games and simulations), 'microworlds' (see below), and general purpose software (such as spreadsheets).

Goldenberg (1982) describes a microworld as, "a well-defined, but limited environment in which interesting things happen and in which there are important ideas to be learned" (p. 218). In a review of the different ways in which the term 'microworld' has come to be used, Edwards (1998) proposes that a microworld can be seen as 'embodying' a sub-domain of mathematics (or science), because of the opportunity that such an environment provides "for students to kinaesthetically and intellectually interact with the designers' construction of these entities, as mediated through the symbol system of a computer program" (p. 74). Examples of relevant research in mathematics education include Christou et al. (2007), Confrey et al. (2009) and Hewitt (2016).

With this in mind, a key question for mathematics education research is how digital environments in mathematics education can be "designed so as to capture significant moments of learning" (CIEM/ICMI 2005, p. 356). Despite the fact that the teaching and learning of proof is recognized internationally as a key component of mathematics education (e.g. Hanna and de Villiers 2008, 2012), numerous studies have shown that students at the secondary school level (and beyond) have serious difficulties in learning about proof and proving (Stylianides et al. 2016). For example, studies assert that students cannot construct deductive proofs (e.g. Senk 1989; McCrone and Martin 2009) or are unable to use existing knowledge strategically (e.g. Weber 2001). There is a tendency for students to rely overly on empirical data or concrete examples (e.g. Harel and Sowder 2007; Küchemann and Hoyles 2006; Martinez and Pedemonte 2014). Even if they can construct a more general argument, they may still verify its correctness via empirical data (e.g. Harel and Sowder 2007) and when asked to produce a proof they may offer an empirical-numerical proof (Stylianou et al. 2015).

To date, digital environments aiming at facilitating learning about mathematical proof and proving might form two categories. One has been developed mainly by researchers in Artificial Intelligence (AI), offering environments that focus on what might be considered more formal aspects of proving. A prototypical example of this is the Geometry Tutor (Anderson et al. 1986), later incorporated in the Cognitive Tutor (Anderson et al. 1995). According to Balacheff and Kaput (1996, p. 478), the Geometry Tutor, at least in its early incarnations, required a student to "construct the proof tree of

some statement [… while] limiting the learner to a linear top-down or bottom-up construction [… even if] providing immediate feedback on errors. [This, they go on to say] keeps the learner away from the problem-solving activity which is tentative, non-linear and involves conjecturing as much as proving" (p. 478).

The second category is characterized by the subsequent development of dynamic geometry environments (DGEs, such as Cabri-Géomètre, The Geometer's Sketchpad and, subsequently, GeoGebra) that have, in contrast to AI proof environments, contributed to stimulating the use of conjecturing and the dialectical relationship between proofs and refutations in mathematics classrooms (e.g. González and Herbst 2009; Komatsu 2017). Systems that can bridge aspects of AI proof environments and DGE environments is a promising area for technology-based development (Botana et al. 2015).

In particular, it is necessary to develop suitable technology-integrated environments, which may help to support the introductory learning of deductive proofs in order to enhance the capabilities of junior high school students in relation to deductive proving (from around the age of 14). This article reports on the pedagogical and technological underpinnings of a web-based learning support system (available in Japanese, English and Chinese) for students who are just starting to tackle deductive proving in geometry.[1] In this article, we explain the design of our web-based system and illustrate, based on initial classroom trialling, how the way in which it is designed can support students' proof construction processes and promote deductive thinking.

## Theoretical Notions and Pedagogical Ideas Underpinning the Design Principles of the Web-based Learning System

The design of our system is based on two theoretical aspects of understanding of geometrical proofs: 'The structure of a proof from a logical point of view' and 'Levels of understanding of the structure of a proof', in addition to the pedagogical idea of 'Exploratory proofs'. We have provided more detailed accounts of these theoretical and pedagogical aspects in Miyazaki et al. (2015, 2017) and in Miyazaki and Fujita (2015). Here, we only briefly explain each of them and then summarize the design principles that informed our web-based learning system.

### Structure of Proofs from a Logical Point of View

A proof generally consists of deductive reasoning starting from premises and leading to conclusions. Within this reasoning process, at least two types of deductive reasoning are employed: *universal instantiation* (which deduces a singular proposition from a universal proposition) and *hypothetical syllogism* (which connects singular propositions logically to provide a bridge between the premises, the intermediate propositions and the conclusion). Considering this, we specify the 'structure of a proof' as the relational network that combines singular and universal propositions by means of these two types of deductive reasoning.

---

[1] The system (with twenty tasks) that can be accessed online at: http://www.schoolmath.jp/flowchart_en/home.html. Five tasks have recently been programmed with html5 and can be accessed at: http://www.sun-first.jp/fc_html5d/.

Figure 1 illustrates a geometrical proof for a proposition that is commonly found in junior high school mathematics, namely that two angles in any isosceles triangle are congruent. More specifically, in ΔABC, if AB = AC, then ∠ABD = ∠ACD. The proof could be established by using universal instantiations and hypothetical syllogisms to demonstrate.

## Levels of Understanding of the Structure of a Proof

In order to understand the structure of proof, students need to pay attention to its elements and their inter-relationships. Research studies by Heinze and Reiss (2004) and by McCrone and Martin (2009) have identified that an appreciation of proof structure is an important component of learners' proof competence.

One of the existing models is the *Reading comprehension of geometry proofs* (RCGP) (Lin and Yang 2007; Yang and Lin 2008), which reflects a local developmental perspective and provides meaningful insight into how the reading comprehension of geometric proofs might advance. On the other hand, the model has limits when applied to proof *construction* due to its original focus on reading comprehension. Moreover, as argued in the previous subsection, geometrical proof is constructed with the combination of two kinds of deductive reasoning: universal instantiation and hypothetical syllogism. The RCGP, however, does not pay detailed attention to the level of 'chaining elements'.

Therefore, in this paper, we use the following levels of learner understanding of proof structure initially elaborated by Miyazaki et al. (2015, 2017): pre-, partial- and holistic-structural levels. These levels are described in Table 1 and the overall
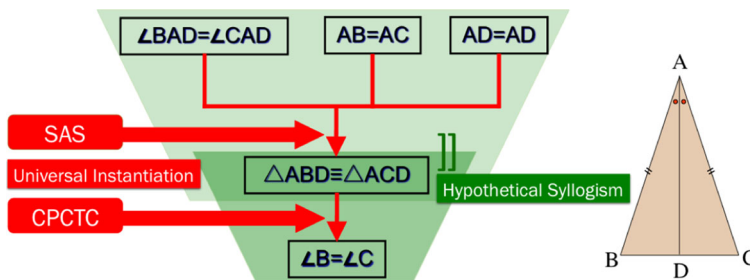


A singular proposition, (i), 'If AB=AC, AD=AD, and ∠BAD=∠CAD, then ΔABD≡ΔACD', is deduced by universal instantiation of the congruency theorem (SAS, a universal proposition)

A singular proposition, (ii), 'if ΔABD≡ΔACD then ∠ABD=∠ACD', is deduced by universal instantiation of the universal proposition 'In congruent triangles all corresponding interior angles are equal' (CPCTC);

These two propositions (i) and (ii) are connected by a hypothetical syllogism, and we obtain 'If AB=AC, AD=AD, and ∠BAD=∠CAD, then ∠ABD=∠ACD', which is equivalent to the singular proposition to be proved.

Fig. 1 Proof of 'the base angles of isosceles triangles are equal'

framework is illustrated in Fig. 2. In our system, this framework is used to determine in what types of feedback can be provided in order to support students' higher levels of understanding of proof. (Specific feedback in terms of types of errors is described towards the end of our article.)

## Exploratory Proving in School Mathematics

Tasks involving proving in mathematics are not limited to writing a proof, but also involve producing propositions inductively/ deductively/ analogically, planning and constructing proofs, looking back over proving processes and overcoming global/local counter-examples or errors, and so on. Based on this broader view, Miyazaki and Fujita (2015) define exploratory proving in school mathematics as having the following three components and interactions among them: producing propositions, producing proofs (planning and construction) and looking back (examining, improving and advancing) (Fig. 3).

Under this idea, students are expected to produce mathematical propositions, produce their proofs as they plan and construct proofs, and then examine the produced proofs, improve them if necessary, and/or advance them to further proving. Through activating the process of explorative proving the transitions of levels of understanding the structure of proof can be enhanced substantially.

## Design Principles to Develop the Proof Learning Support System

As stated at the outset, we are aware that the learning about proofs is not an easy task for many students and it is necessary for research to consider how learners can be supported so that more students can engage in exploratory proving. By considering how an effective system can be developed based on the above theoretical underpinnings, the following points of our design principles seem important for supporting learners who have not much experience in exploratory proving in school mathematics.

First is the necessity for students to be able to visualize the structure of proofs so that they can understand what elements and relations between them are necessary and how they function deductively within them. This enriches their understanding of the

**Table 1** Levels of understanding of the structure of a proof

| Level | Description |
|---|---|
| *pre-structural* | The basic status in terms of an understanding of proof structure where learners regard proof as a kind of 'cluster' of possibly symbolic objects. |
| *partial-structural* | Once learners have begun paying attention to each element, then we consider they are at the *partial-structural elemental* sub-level. To reach the next sub-level, learners need to recognize some relationships among these elements (such as universal instantiations and syllogism). If learners have started paying attention to each relationship, then we consider them to be at the *partial-structural relational* sub-level, with this sub-level being further divided into (a) universal instantiation and (b) syllogism (see Fig. 2). |
| *holistic-structural* | At this level, learners understand the relationships between singular and universal propositions, and can see a proof as a 'whole', one in which premises and conclusions are logically connected through universal instantiation and hypothetical syllogism. |
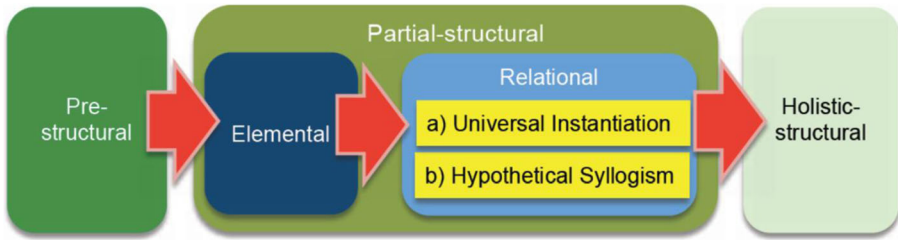
**Fig. 2** Framework of learner understanding of the structure of a proof

structure of proofs, especially from the partial-structural elemental sub-level to the relational sub-level and to the holistic-structural level. Under this principle, the design of a learning support system of proof allows approval of the presentation of the whole form of a flow-chart proof with blanks to support the introductory learning of formal proofs as shown the next section.

The second principle is the realization of learning opportunities for explorative proving so that students can plan proofs together with constructing them, examine their proof construction processes/product, improve them if necessary, and/or advance them to further proving. Under this principle, focusing on planning proofs as the essential part of explorative proving, the design of the system get to adopt the usage of open-problem situation of proving to amplify the planning proofs as shown in the second subsection below. This usage can enhance students' capacity to think backward and forward to connect conclusions with assumptions. Additionally, the principles of feedback (see final subsection below) are carefully considered so that learners can modify and improve their proving activities, which we consider realizes the interaction between 'Producing proof' and 'Looking back' as the critical parts of explorative proving in school mathematics.
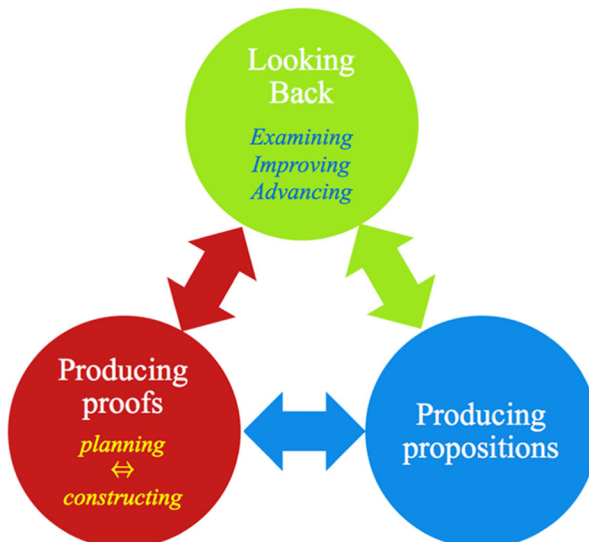


**Fig. 3** Exploratory proving in school mathematics

# Design of a Web-based Learning Support System for Flow-chart Proving in School Geometry

A format of flow-chart proof (see, for example, McMurray 1978) pictures the deductive connections from premises to conclusion by distinguishing singular and universal propositions in the diagram. The flow-chart format proof can be adopted before using other proof formats such as paragraph proofs or two-column proofs (Gardiner 2004). Critically, the representational feature of flow-chart proof can take the role of mediational tools between natural language and functional language (Balacheff 1987) as proofs demand to make full use of the peculiar symbolic systems.

In the introductory teaching of proof, it is particularly important to support the transition from the Partial-structural Elemental sub-level to Relational and to the Holistic-structural level. Therefore, we use the whole form of flow-chart proof with blanks to aim to help students visualize that a deductive proof consists of two kinds of propositional layers, one of which contains universal propositions (theorems) and the other contains the chain of singular propositions. Also, students can see clearly that a singular proposition is deduced by the universal instantiation of universal proposition, and that the chain of singular propositions between assumptions and conclusions would be established by hypothetical syllogism (Miyazaki et al. 2015). As a result, we can expect that students can be encouraged to develop towards the Partial-structural Relational sub-level and toward the Holistic-structural level of the structure of deductive proofs.

Some existing learning support systems adopt a diagrammatic format of proof. For example, *Geometry Tutor* (Anderson et al. 1986), and *Cognitive Tutor* (Anderson et al. 1995) make use of a 'flow chart tool' by which learners construct proofs step-by-step by choosing appropriate elements of proofs.

Our system uses a similar idea. The flow-chart format is used in a way designed for supporting proof learning. Our system is also designed in order fully to support the development of the understanding of the structure of deductive proofs, i.e. we consider the possibility that the above difficulties of constructing proofs (e.g. problems in connecting premises, intermediate propositions and conclusion) might be caused by insufficient student understanding of the structure of proofs (see Table 1). As above, the system shows the whole form of flow-chart proof with the blanks from the beginning (see Fig. 4).

This visualization scaffolds learners to grasp the whole structure of the proof which they are about to construct with the aim to enhance their understanding of the general structure of geometric proofs. In contrast, with the Cognitive Tutor not all the figural theorems used in proofs are visually represented (except for the theorems of triangle congruency) (Anderson et al. 1995, p. 173). In the recent version of Cognitive Tutor, the theorems used to deduce the figural relations are hidden inside the boxes of singular propositions after learners choose the relations to give the reason why.[2] In our system, as shown in Fig. 4, all of the used figural theorems are always visible with 'tabs' in order to support students' understanding related to universal instantiations as proof structures.
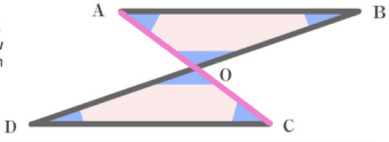
---

[2] As an example, see: http://www.schooltube.com/video/08dde7e83fd743323451/GEO-63-Proofs-Using-Parallel-Lines-Theorems.

**Fig. 4** Showing flow-chart format with blanks from scratch

## Usage of Open-Problem Situation of Proving to Amplify the Planning of Proofs

We propose that the use of flow-chart proofs might be enhanced at the introductory stage of proof learning by using 'open problem' situations where students can construct multiple solutions by deciding the assumptions and intermediate propositions necessary to deduce a given conclusion.

For example, the problem in Fig. 4 is intentionally designed so that students can freely choose which assumptions they use to prove that AB = CD. Students might decide, for instance, that ΔABO and ΔCDO should be congruent to show AB = CD by using the theorems "If two figures are congruent, then corresponding sides are equal." Based on OA = OC as an assumption, ΔABO ≡ ΔCDO can be shown by assuming BO = DO and ∠BOA = ∠DOC using the SAS condition. However, other solutions are also possible. One approach might be to use the fact that ΔABO ≡ ΔCDO can be shown by assuming OA = OC, ∠BOA = ∠DOC and ∠OAB = ∠OCD, using the ASA condition for congruency. As students can construct more than one suitable proof, we refer to this type of problem situation as 'open'.

Compared to the other existing proof support systems, we cannot find anything which uses open situations of proving. We consider this is an important and essential idea for developing students' understanding of the structure of proof in geometry. In accordance with our theoretical framework of explorative proving in school mathematics, this usage of open situations would encourage learners to seek out the necessary assumptions and intermediate propositions by thinking backward and forward productively. As a result, with open situations they are expected to be able to plan and evaluate proofs more effectively when constructing proofs (for more on this, see Miyazaki et al. 2015).

## Providing Systematic Feedback during/after Constructing Proofs

Our system starts systematically giving forms of feedback after students complete their tentative proofs by responding to four types of errors described below and the

principles of the order of feedback that follow, based on the proof structure described above. In accordance with our theoretical framework of explorative proving in school mathematics, these systematic forms of feedback aim at encouraging learners to examine and improve their proof successively. Through interacting with the system, learners are offered opportunities to acquire the habit of checking their proofs and find ways to examine and improve their incorrect proofs towards deductively correct proofs.

Students are likely to make various errors when using our system. It is challenging to know how to provide formative feedback in order to support learning (Sangwin et al. 2009), but technology can aid the provision of timely and targeted feedback (Panero and Aldon 2016). In our system, learners' answers are referred to a database of correct answer. If there is an error, the system recognizes it, and then gives a feedback message. In order to encourage learners' self-reflective skills, the system uses our theoretical underpinnings of the structure of proof. Errors are classified into the following four categories: Category A is related to hypothetical syllogism, Category B to universal instantiation, Category C to singular proposition, and Category D to the syntax of deductive proof.

### Category A: Errors Related to Hypothetical Syllogism

The errors classified into this category are caused by lack of understanding of hypothetical syllogism necessary to make a chain of singular proposition between assumptions and conclusions in a proof (i.e. such students are considered to be at the Partial structural Elemental level; see Table 1 and Fig. 2). By the treatment and improvement of this type of error, learners might be encouraged to reflect on the logical relationships between premises and conclusions, and this might lead to an enrichment of their understanding of hypothetical syllogisms, which is necessary to reach the Partial structural Relational level b) or the Holistic level of understanding.

Figure 5 shows an example of the system giving feedback that the proof falls into logical circularity. The conclusion 'AB = CD' is used as the one of three condition to deduce the congruence of triangle 'ΔABO ≡ ΔCDO' despite of the correct usage of each theorem. Therefore, the system shows a message 'You cannot use the condition to prove your conclusion!' By referring to this message the learner is expected to examine the condition used in the (incorrect) proof, and replace it with, for instance, the included angle'∠AOB = ∠COD' accompanied by the re-selection of the SAS condition of congruent triangles.
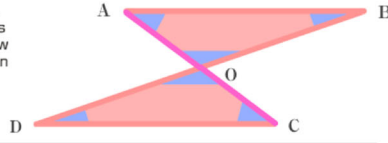
### Category B: Errors Related to Universal Instantiation

The errors classified into this category are caused by misusing universal instantiation which is necessary to produce singular propositions from universal propositions (theorems, definitions, and so on; i.e. such students are considered to be at the Partial structural Elemental level, Table 1 and Fig. 2). By the treatment and improvement of this type of error, learners might be encouraged to reflect on what universal propositions should be used to deduce logical relationships of singular propositions, which is necessary to reach the Partial structural Relational level a) or the Holistic level of understanding.
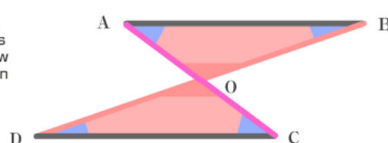
Fig. 5 Feedback for logical circularity

Figure 6 shows that the system identified that an incorrect congruent condition of triangles (universal proposition) was used. In order to deduce the congruence of triangle 'ΔABO ≡ ΔCDO' by using three singular propositions ('OA = OC', '∠AOB = ∠COD', 'BO = DO'), the congruent condition of triangles should not be SSS but SAS. The system shows the message 'You may select again a condition of congruence of two triangles'. By referring to this feedback, the learner is expected to examine the congruent condition and replace SSS with SAS.



Fig. 6 Feedback for the use of universal proposition

*Category C: Errors Related to Singular Propositions*

The errors classified into this category are caused by making mistakes in constructing singular propositions, which indicates such students might be at the Pre-structural level of understanding (see Table 1 and Fig. 2). By the treatment and improvement of this type of error, students would be encouraged to consider what singular propositions should be used in order to construct correct proofs, which is necessary for the Partial structural Elemental level of understanding.

Figure 7 shows that the system identifies mistakes related to singular propositions. In order to deduce the congruence of triangle '△ABO ≡ △CDO" by using SAS condition of congruent triangle and two singular propositions ('AO = CO', 'BO = DO'), the remained singular proposition should not be '∠ABO = ∠CDO' but '∠AOB = ∠COD'. Therefore, the system shows the message 'Let us find the included angle of these sides'. By referring to this feedback, the learner is expected to find the included angles between two sides and replace '∠ABO = ∠CDO' with '∠AOB = ∠COD'.

*Category D: Errors Related to Proof-Format*
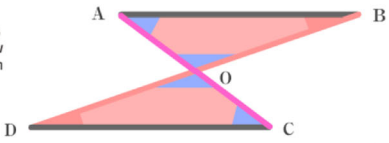
The errors classified into this category are caused by proof-format errors, usually rather trivial ones related to singular propositions. In general, deductive proofs should be constructed such that singular proposition concerning angles or sides corresponds to a singular proposition concerning triangle congruency.

Figure 8 shows an example of the system identifying a proof-format mistake. In order to keep the proper arrangement of each sentence according to the conclusion 'AB = CD', the singular proposition '∠COD = ∠AOB' should be changed to '∠AOB = ∠COD'. Therefore, the system shows the message 'Be careful of the



**Fig. 7** Feedback for necessary conditions

**Fig. 8** The order of comparisons is checked with a message

order of comparisons!'. By referring to this feedback the learner is expected to swap the left and the right sides of the singular proposition '∠COD = ∠AOB'.

### Principles of the Order of Feedback Based on the Structure of Proof

It is expected that a learner might make more than one type of error simultaneously (e.g. both type A and type B errors). As such, an issue to be considered is in what order the feedback should be given so that the system can support students. Here, we take extra care of the order of which error category should be resolved.

From the viewpoint of proof structure shown in the previous section, our system database searches for Type A first, and feedback related to this type of error is prioritized over the other types of error (B, C and D). Given that a proof locally connects premises and conclusion, and consists of universal propositions (such as theorems) and singular propositions (such as properties of geometrical figures specified for each proof), this logical connection is broken by logical circularities. Our system prioritizes notifying this type of error, which we call type A, first and foremost. Elements of proofs such as singular propositions depend on what universal positions are being used. As such, our system takes the type B errors more seriously than the types C and D. As type D errors are essentially 'how to write a proof more accurately and formally', then this type of error is not considered as the most important.

The system follows the principle of three rules in order, i.e. Rule 1 → 2 → 3:

- Rule 1: Category A > Category (B, C, D): The system takes the errors related to hypothetical syllogism as the most important error; this is because a proof with logical circularity is meaningless as a whole.
- Rule 2: Category B > Category C: Errors in category B are then corrected, in order to support students' understanding of the use of universal propositions.

- Rule 3: Category C > Category D: Finally, the mistakes related to singular propositions (Category C) are taken ahead of proof-format error (Category D) from the logical point of view.

These rules echo the level of understanding of proof structure shown in the previous section. By correcting errors according to these rules, learners can be expected to shift their understanding of proof structure gradually, and acquire the appropriate means to find and improve their errors according the levels of understanding of proof structure.

Nevertheless, when there are many universal and singular propositions in a proof, it is necessary to decide the order of errors that should be resolved in the same category. In order to enhance learners' capacity to construct deductive proofs they need to think backwards from conclusions to assumptions, rather than only think forwards from assumptions to conclusions. Then, the system requests learners to tackle their errors in turn from conclusions to assumptions. Thus, the system controls the order of error improvement in both directions from reasoning to propositions and from conclusions to assumptions. The former improvement would enhance the understanding of proof structure, and the latter would strengthen the capacity to construct deductive proofs.

While automatic feedback is one of the central features of our system, we consider that the teacher's role is also crucial in enabling learners to learn the construction of proof in geometry. This is in line with Mariotti's (2000) findings that support from the artefact (in our case, our system) can be enhanced by the teacher's guidance and that the latter should not be underestimated. The artefact incorporates mathematical knowledge accessible to the learner through its design and use, but the construction of meaning requires the guidance of the teacher who organizes and directs specific activities in which the development of meanings can be recognized and accepted mathematically.

## Production of the web-Based Learning Support System

Informed by the principles underpinning the design of the web-based learning support system described in the previous section, the following are three technological features that we utilise to support students' learning of proofs within our system of flow-chart proving:

- interface to translate automatically figural to symbolic objects;
- reviewing learner's previous correct answers;
- organizing learner tasks according to the complexity of problems.

In the next sub-section, we explain how these technological features are used in our system.

### Interface to Translate Automatically Figural to Symbolic Objects

One of the main difficulties students face in constructing proofs is expressing the figural relations (e.g. $\angle BAO = \angle CAO$) and the theorems (e.g. SSS "If three pairs
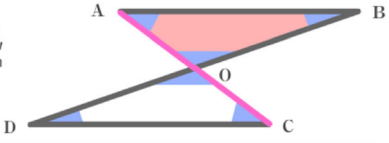
of sides of two triangles are equal in length, then these triangles are congruent"). For students in the introductory stages of learning of deductive proofs, writing down the figural relations with letters (symbols) is a demanding task for Japanese students (Koseki, 1987). To reduce this representational barrier, our system converts the figural parts into symbolic representation when relevant parts of diagram are 'dragged' into the boxes of the flow-chart proof.

For example, to insert the symbol ΔCDO in the target box of flow-chart proof in the problem in Fig. 9, the triangle in the diagram can be 'grabbed' and then to dragged and dropped it into the box. Then, the symbol ΔCDO appears inside the box. There is no need to type the letters into the box.
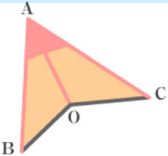


Fig. 9 Replacing a triangle with the symbol ΔCDO

## Reviewing a Learner's Correct Answers

Our tasks that use an open-problem situation have multiple solutions. Some of our tasks have two, three or four different solutions, with one task having eight different solutions. In order to complete these kinds of tasks, learners are expected to seek multiple solutions, and, in such processes, are expected to find it useful to review the solutions that they have already found.

The task shown in Fig. 10 has four kinds of solutions indicated with the number of stars ☆. If learners find some solutions, the stars corresponding to these solutions are filled in yellow. In the case illustrated, the learners have already found two solutions (shown by the two yellow stars in Fig. 10). As part of their search for the two remaining solutions, they can review their previous answers by clicking each yellow star (an example solution is shown in the lower part of Fig. 10).



Fig. 10 A task with multiple solutions and a learner solution

The facility for learners to review their existing answers is designed to encourage learners to consider other proofs. For example, the proof shown in the upper part of Fig. 10 uses 'AB = AC', '∠OAB = ∠OAC' and 'OA = OA', but this proof can be changed into a new proof by replacing all or some of the assumptions. The lower part of Fig. 10 shows a proof with SSS. In considering further proofs, learners might seek proofs with either SAS or ASA conditions, by replacing, for example, ∠OAB = ∠OAC with ∠BOA = ∠COA and AB = AC with BO = CO while keeping SAS condition. If the SAS condition is changed to ASA, learners can produce another proof (making four in total). However, if they choose ∠ABO = ∠ACO as an assumption, they might fall into logical circularity. This latter case can reveal learners' uncertainty about the structure of proofs; see the final section below (and also Miyazaki et al. 2017).

## Organizing 'Lessons' According to 'Goal' and 'Complexity'

In order to support the introductory stages of proof learning, it is necessary to organize the content of simple proofs according to the difficulty of the problem situations. As shown in Fig. 11, our web-based system has a front page which organizes 20 'lessons' (each comprising one task). This organization of 'lessons' has two directions: 'Goal' and 'Complexity'.

Goals are classified into five 'levels'. LEVEL 5 is the highest level and assumes that learners can master how to prove basic properties of triangles and parallelogram by using the properties of parallel lines, the conditions for congruence of triangles, and so on. The content of LEVEL 5 requires learners to construct proofs with more than two steps of deductive reasoning. Before reaching LEVEL 5, learners should master how to use the properties of parallel lines and angles (LEVEL 1) and to make two triangles congruent with the conditions for congruence of triangles (LEVEL 2) as the basic skills. These levels require one-step deductive reasoning.



Fig. 11 Organization of twenty 'lessons'

In order to construct proofs with more than two steps of reasoning, learners should master how to make two sides/angles equal using the conditions for congruence of triangles and the properties of congruent figures (LEVEL 3). In addition, they need to master how to make two triangles congruent using the properties of parallel lines and the conditions for congruence of triangles (LEVEL 4).

The detailed order of levels was originally motivated to improve the teaching of proof in Japan and is therefore strongly connected with the Japanese school curriculum in geometry. Nevertheless, the general progression from open to close proof problems with flow-chart format might be applicable for any geometry curricula which include the teaching of deductive proofs.

Open problem situations are used across LEVEL 1 to 4 in order to enhance learners' capacity to think backwards from conclusions to assumptions, something that contributes in LEVEL 5 to planning and constructing proofs with more than two steps of reasoning in closed problem situations. Table 2 summarizes the characteristics of each level; the steps of reasoning, open or close situations, and the number of 'lessons'.

In Fig. 11, the degree of complexity of contents is shown from left to right. The complexity of the four or five 'lessons' per level are arranged according to the difficulty of each problem situation. There are three categories: Introduction; Practice; and Advanced. In LEVEL 3, for example, Lesson III-1 is an 'Introduction' that provides the situation (shown in Fig. 9) where two triangles sharing only one vertex should be congruent, Lesson III-2, as 'Practice', provides the situation where two triangles sharing one side should be congruent (see Fig. 10), and Lesson III-3, as 'Advanced,' provides further challenge for learners in the form of a problem where two overlapping triangles should be congruent (see Fig. 12).

## Learner Support Provided by the Web-based System

From our initial classroom trialling, we can expect that the system provides two kinds of supports for the learning of proofs. One is to understand the structure of proof and
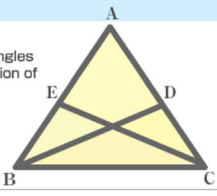
**Table 2** Characteristics of five levels

| Levels | Goals | Steps of reasoning | Situations | Number of 'lessons' |
|---|---|---|---|---|
| LEVEL 5 | Master how to prove basic properties of triangles & parallelogram | 3 | Closed | 1 |
| | | 2 | | 4 |
| LEVEL 4 | Master how to make two triangles congruent with the properties of parallel lines and the conditions for congruence of triangles | 2 | Open | 4 |
| LEVEL 3 | Master how to make two sides/angle equal with the conditions for congruence of triangles and the properties of congruent figures | 2 | Open | 4 |
| LEVEL 2 | Master how to make two triangles congruent with the conditions for congruence of triangles | 1 | Open | 4 |
| LEVEL 1 | Master how to use the properties of parallel lines and angles | 1 | Open | 3 |

**Fig. 12** Problem involving two overlapping triangles

the other is to plan deductive proofs. Here, amongst the data from our pilot study, we demonstrate these two kinds of support through the case of a pair of UK undergraduate trainee primary school teachers, J1 & J2 (each of whom had mathematics as their specialism but limited experience of deductive proofs in geometry), who used our system of proof problems from LEVELS 2 to 5. Their working was video-recorded by a video camera and screen shots. We acknowledge we show only one case but as we illustrate, their learning experiences include many interesting points in relation to the status of their understanding and development of deductive proofs in geometry.

### Learner Support to Understand the Structure of Proof

Concerning the support to amplify the understanding of the structure of proof, using the system in the introductory lessons of deductive proofs is designed to enable learners to make singular propositions (e.g. 'AB = CD') by dragging side/angle/triangle etc. of a diagram into the flow-chart proof, and the system checks the produced propositions and, if necessary, provides learners with an informative suggestion. For example, the case of J1 and J2 who were undertaking one of the problems at LEVEL 2 (see Fig. 13 and Table 2) exemplified the understanding of universal instantiation as follows:

24.  J1 And then side-angle-angle?
25.  J2 [completing a proof]
26.  J1&J2 [choosing one of the conditions]
27.  J1 We are going to have to choose the middle ones…
28.  J2 Yes. [changing the angles: Angle BEO to BOE]
29.  J1&J2 [checking the answer] Yes!
30.  J1 Um, it is… try once more. Which… you know we did 'side-angle-side'
31.  J2 [Clicking one of the starts]
32.  J1 That is 'side-angle-angle', I think the left one.

Fig. 13 J1 and J2 tackle a LEVEL 2 proof problem

33. J1 So we used DAO… now, middle angles (AOD and BOE)?
34. J1&J2: Yes!

Here, the learners could correctly choose the appropriate theorems (universal propositions) from the pull-down list in order to establish the deductive relation of singular propositions in the flow-chart proof format. At this point, they became aware that they could find alternative solutions (e.g. turn 27 J1 "We are going to have to choose the middle ones …" or see turns 30–34). The system format visualizes the difference of singular propositions and universal propositions. Through choosing the theorems from the list, learners can discriminate two kinds of propositions along with the structure of proof. This discrimination is essential for the Partial-structural Relational (a) universal instantiation) sub-level.

Fig. 14 J1 and J2 tackle a LEVEL 3 proof problem

Furthermore, if a proof falls in logical circularity the system firstly specifies the error as Category A, and requests learners to improve with the supportive message and the marks on screen (see Fig. 5). Through the improvement, learners can understand that they should not use conclusions as assumptions. This understanding is essential for the Partial-structural Relational (b) hypothetical syllogism) sub-level (Miyazaki et al. 2017). In fact, both J1 and J2, after they experienced LEVELS 2, 3 and 4 and received various types of feedback from the system, came to know that in a proof they cannot use the same singular proposition as both assumption and conclusion. For example, in the extract in Fig. 14, J2, in line 212, suggested that the ASA condition and AB-angle ABO-BO be used but J1 challenged that they cannot use angle ABO (see lines 212–218). As a result, they then found a correct proof.

212. J2 [Reading the tab] We can try ASA? Because if we are showing these figures are congruent, you can still reach… Wait, we are concluding about angles. So what we use is different sides… If we use this one [AB] and this one [BO] then we can use [ABO]
213. J1 You can't use these angles [ABO] because these angles we find…
214. J2 Yes, then, this one? Wait then…
215. J1 That angle there [AOB] and those two sides [AO and OB]
216. J2 So we have to pick, this one, two sides [SAS]. And then we are using this one [AO] and this one [OB].
217. J1 Yes.
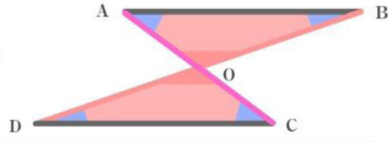218. J2 These two are equal [AO] and then this one and this one [AOB and AOC]. Yes? [laugh]

## Support to Plan a Proof

Concerning the enhancement of the capacity to plan and construct deductive proofs, the system monitors for possible multiple errors in a proof along both directions from

**Fig. 15** J1 and J2 tackle another LEVEL 3 proof problem

universal to singular propositions and from conclusions to assumptions (using the principles of the order of feedback based on the structure of proof). In particular, examination of the latter direction promotes thinking backwards that learners should bear in mind when planning and constructing proofs.

For example, in tackling the range of 'lessons' from LEVEL 2 to 5, J1 and J2 often started choosing the conditions of the congruent triangles and then changed assumptions, as illustrated in Fig. 15.

20. J1 [after completing one of proofs for the lesson III-1] Check your answer…
21. J1&J2 What?
22. J1 [reading the message on the screen] "Be careful of the order of the comparison!"… Oh, I've got the triangles wrong way around! So that one [OAB] and that one [OCD]. Maybe? Yes! Because I put everything from that top triangles in left… and then I just switched order… I see.
23. J2 Try once more…
24. J1 So shall we use 'angle-side-angle'? We could do two angles?
25. J2 Yes.
26. J1 So they have to be those two [triangle OAB & AOB], so these are included sides…
27. J2 Right.
28. J1 So, this one [AOB] and this one [COD], and then this one [OAB] and this one [DCO]. Now … [choosing one of the conditions]. And then this triangle [AOB] first… equals this triangle [COD].
29. J2 Then the last box… and then it's proven?
30. J1 Yes.

After responding to the message from the system (turns 22–23) by correcting the triangle correspondence (turns 22–23), J1 and J2 discussed how to deduce the

congruency 'triangle ABO = CDO' by thinking backwards. They then planned to find two pairs of angles ('angle AOB = COD' and 'angle BAO = DCO') that would make it possible to use the assumption 'AO = CO' (turns 24–28). Thus, both J1 and J2 were planning their proofs by considering which condition of congruent triangles could be used before they started constructing their proof.

## Coda: Limitations and Prospects

The cases above illustrate our findings from our pilot study that our web-based system can support the learning of deductive proof (both from universal to singular propositions, and from conclusions to assumptions). Nevertheless, we are aware that there may be some limitations in terms of learners' skills in forming mathematical expression. For example, in using the interface to replace automatically the figural with the symbolic, learners may not experience sufficient opportunity to practice expressing propositions using symbols and characters (e.g. expressing '$\angle COD = \angle AOB$' in symbols).

More seriously, while using the open flow-chart proof format may support learners to construct deductive proofs, the format is not in regular use in classrooms. As a next step after using our system, learners might need to learn how to express their proof in a more conventional style such as a paragraph proof. As such, it may be necessary to provide lessons for students to learn how to re-write flow-chart proofs into paragraph proofs.

In order to exploit the positive supports and minimise the limitations noted above, it is necessary to develop a learning progression for the introductory lessons of deductive proofs with the use of the system. In earlier research, we proposed and examined the progression with open/closed flow-chart proving and its effectiveness as scaffolding for learning deductive proof (Miyazaki et al. 2015) for the introductory lessons of formal proofs without the web-based system as follows: (1) constructing flow-chart proofs with open problems; (2) constructing formal proofs with closed problems with reference to flow-chart proofs; (3) refining formal proofs with closed problems by placing them into a flow-chart proof format. Based on this progression, we have already developed a guidance book for teachers on how to manage the introductory lessons of proof that includes the contents of the system (although, at the moment, the teacher guide is only in Japanese; http://www.schoolmath.jp). Future research might examine how these lessons, along with the progressions with the system, function in a range of classrooms.

Beyond the proposed learning progressions, future research could develop, implement and evaluate a geometry curriculum that includes proofs with the use of the web-based system. At the moment, we have already developed and implemented a curriculum for explorative proving at the junior high school level in Japan without use of the web-based system (Miyazaki et al. 2016). As a next step, there is need to elaborate the curriculum with the web-based system. For this development and research, the quality of teachers' instructions should be seriously considered. For example, a study of the use of the Cognitive Tutor found a negative impact on students' achievement in geometrical proofs, arguably because teachers using the system had difficulties in implementing the approach (Pane et al. 2010). This suggests that in order to examine the effect of our web-based system, careful designs for effective teaching approaches with the system and rigorous evaluations would be indispensable.

For out-of-school lessons the web-based system could be used to enable the chance to learn deductive proof at home. Without teacher's guidance, however, it can be difficult for home-learners to use the system effectively. By introducing gaming elements to the system, learners might enjoy the challenge of constructing flow-chart proofs. Research into incorporating the notion of gamification (Karl 2012) to the system could contribute to the further productive use of the system.

# References

Anderson, J., Boyle, C., & Yost, G. (1986). Using computers to teach: The geometry tutor. *The Journal of Mathematical Behavior, 5*(1), 5–19.

Anderson, J., Corbett, A., Koedinger, K., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences, 4*(2), 167–207.

Balacheff, N. (1987). Processus de preuve et situations de validation. *Educational Studies in Mathematics, 18*(2), 147–176.

Balacheff, N., & Kaput, J. (1996). Computer-based learning environments in mathematics. In A. Bishop, M. Clements, C. Keitel, J. Kilpatrick, & C. Laborde (Eds.), *International handbook of mathematics education* (pp. 469–501). Dordrecht: The Netherlands: Kluwer Academic Publishers.

Botana, F., Hohenwarter, M., Janičić, P., Kovács, Z., Petrović, I., Recio, T., & Weitzhofer, S. (2015). Automated theorem proving in GeoGebra: Current achievements. *Journal of Automated Reasoning, 55*(1), 39–59.

Brown, G., Cadman, K., Cain, D., Clark-Jeavons, A., Fentem, R., Foster, A., Jones, K., Oldknow, A., Taylor, R. and Wright, D. (2005, revised edition). *ICT and mathematics: A guide to learning and teaching mathematics 11–19*. Leicester, UK: Mathematical association.

Christou, C., Pittalis, M., Mousoulides, N., & Jones, K. (2007). Developing the 3DMath dynamic geometry software: Theoretical perspectives on design. *The International Journal for Technology in Mathematics Education, 13*(4), 168–174.

Commission Internationale de l'Enseignement Mathématique/International Commission on Mathematical Instruction. (2005). Digital technologies and mathematics teaching and learning: Rethinking the terrain (technology revisited). *L'Enseignement Mathématique, 51*(3–4), 351–363.

Confrey, J., Hoyles, C., Jones, D., Kahn, K., Maloney, A., Nguyen, K., Noss, R., & Pratt, D. (2009). Designing software for mathematical engagement through modeling. In C. Hoyles & J.-B. Lagrange (Eds.), *Mathematics education and technology: Rethinking the terrain* (pp. 19–45). Berlin: Germany: Springer.

Edwards, L. (1998). Embodying mathematics and science: Microworlds as representations. *The Journal of Mathematical Behaviour, 17*(1), 53–78.

Gardiner, T. (2004). Learning to prove: Using structured templates for multi-step calculations as an introduc-tion to local deduction. *ZDM: The International Journal on Mathematics Education, 36*(2), 67–76.

Goldenberg, P. (1982). LOGO: A cultural glossary. *Byte, 2*(8), 210–229.

González, G., & Herbst, P. (2009). Students' conceptions of congruency through the use of dynamic geometry software. *International Journal of Computers for Mathematical Learning, 14*(2), 153–182.

Hanna, G., & de Villiers, M. (2008). ICMI study 19: Proof and proving in mathematics education. *ZDM: The International Journal on Mathematics Education, 40*(2), 329–336.

Hanna, G., & de Villiers, M. (2012). Aspects of proof in mathematics education. In G. Hanna & M. de Villiers (Eds.), *Proof and proving in mathematics education: The 19th ICMI study* (pp. 1–10). New York, NY: Springer.

Harel, G., & Sowder, L. (2007). Toward a comprehensive perspective on proof. In F. Lester (Ed.), *Second handbook of research on mathematics teaching and learning* (pp. 805–842). Reston: VA: National Council of Teachers of Mathematics.

Heinze, A. & Reiss, K. (2004). Reasoning and proof: Methodological knowledge as a component of proof competence. In M. Mariotti (Ed.), *Proceedings of CERME 3 (WG4 argumentation and proof)*. Augsburg: European Society for Research in Mathematics Education. (www.mathematik.uni-dortmund. de/˜erme/CERME3/tableofcontents_cerme3.php).

Hewitt, D. (2016). Designing educational software: The case of *Grid Algebra*. Digital Experiences in Mathematics Education, 2(2), 167–198.

Karl, K. (2012). *The gamification of learning and instruction: Game-based methods and strategies for training and education*. San Francisco, CA: Pfeiffer.

Komatsu, K. (2017). Fostering empirical examination after proof construction in secondary school geometry. *Educational Studies in Mathematics*. (Online First).

Koseki, K. (Ed.) (1987). *The teaching of geometrical proof*. Tokyo: Meiji Tosho publisher [in Japanese].

Küchemann, D., & Hoyles, C. (2006). Influences on students' mathematical reasoning and patterns in its development: Insights from a longitudinal study with particular reference to geometry. *International Journal of Science and Mathematics Education, 4*(4), 581–608.

Lin, F., & Yang, K. (2007). The reading comprehension of geometric proofs: The contribution of knowledge and reasoning. *International Journal of Science and Mathematics Education, 5*(4), 729–754.

Mariotti, M. (2000). Introduction to proof: The mediation of a dynamic software environment. *Educational Studies in Mathematics, 44*(1–3), 25–53.

Martinez, M., & Pedemonte, B. (2014). Relationship between inductive arithmetic argumentation and deductive algebraic proof. *Educational Studies in Mathematics, 86*(1), 125–149.

McCrone, S., & Martin, T. (2009). Formal proof in high school geometry: Student perceptions of structure, validity and purpose. In D. Stylianou, M. Blanton, & E. Knuth (Eds.), *Teaching and learning proof across the grades* (pp. 204–221). London: UK: Routledge.

McMurray, R. (1978). Flow proofs in geometry. *The Mathematics Teacher, 71*(7), 592–595.

Miyazaki, M., & Fujita, T. (2015). Proving as an explorative activity in mathematics education: New trends in Japanese research into proof. In B. Sriraman (Ed.), *First sourcebook on Asian research in mathematics education: China, Korea, Singapore, Japan, Malaysia and India* (pp. 1375–1407). Charlotte: NC: Information Age Publishing.

Miyazaki, M., Fujita, T., & Jones, K. (2015). Flow-chart proofs with open problems as scaffolds for learning about geometrical proofs. *ZDM: The International Journal on Mathematics Education, 47*(7), 1211–1224.

Miyazaki, M., Fujita, T., & Jones, K. (2017). Students' understanding of the structure of deductive proof. *Educational Studies in Mathematics, 94*(2), 223–239.

Miyazaki, M., Nagata, J., Chino, K., Fujita, T. Ichikawa, D., Shimizu, S., & Iwanaga, Y. (2016). Developing a curriculum for explorative proving in lower secondary school geometry. *Paper presented at the 13th international congress on mathematical education*. Hamburg, Germany.

Pane, J., McCaffrey, D., Slaughter, M., Steele, J., & Ikemoto, G. (2010). An experiment to evaluate the efficacy of cognitive tutor geometry. *Journal of Research on Educational Effectiveness, 3*(3), 254–281.

Panero, M., & Aldon, G. (2016). How teachers evolve their formative assessment practices when digital tools are involved in the classroom. *Digital Experiences in Mathematics Education, 2*(1), 70–86.

Sangwin, C., Cazes, C., Lee, A., & Wong, K. (2009). Micro-level automatic assessment supported by digital technologies. In C. Hoyles & J.-B. Lagrange (Eds.), *Mathematics education and technology: Rethinking the terrain* (pp. 227–250). Berlin: Germany: Springer.

Senk, S. (1989). Van Hiele levels and achievement in writing geometry proofs. *Journal for Research in Mathematics Education, 20*(3), 309–321.

Stylianides, A., Bieda, K., & Morselli, F. (2016). Proof and argumentation in mathematics education research. In A. Gutiérrez, G. Leder, & P. Boero (Eds.), *The second handbook of research on the psychology of mathematics education* (pp. 315–351). Rotterdam: The Netherlands: Sense Publications.

Stylianou, D., Blanton, M., & Rotou, O. (2015). Undergraduate students' understanding of proof: Relationships between proof conceptions, beliefs, and classroom experiences with learning proof. *International Journal of Research in Undergraduate Mathematics Education, 1*(1), 91–134.

Weber, K. (2001). Student difficulty in constructing proofs: The need for strategic knowledge. *Educational Studies in Mathematics, 48*(1), 101–119.

Yang, K., & Lin, F. (2008). A model of reading comprehension of geometry proof. *Educational Studies in Mathematics, 67*(1), 59–76.