Learning to Argue in Mathematics: Effects of Heuristic Worked Examples and CSCL Scripts on Transactive Argumentation

Freydis Vogel[1], Elisabeth Reichersdorfer[2], Ingo Kollar[1], Stefan Ufer[1], Kristina Reiss[2], Frank Fischer[1]
[1]LMU Munich, Leopoldstr. 13, D-80802 Munich, Germany,
[2]Technische Universität München, TUM School of Education, Schellingstr. 33, D-80799 Munich, Germany,
Email: freydis.vogel@psy.lmu.de, elisabeth.reichersdorfer@tum.de, ingo.kollar@psy.lmu.de,
ufer@math.lmu.de, kristina.reiss@tum.de, frank.fischer@psy.lmu.de

Abstract: A previous study has shown that both CSCL scripts and heuristic worked examples implemented in a CSCL environment were effective to fostering students' acquisition of argumentation skills in the context of mathematical proof tasks (Kollar, et al. 2012). This paper investigates the extent to which transactive argumentation during the collaborative learning process can be evoked by both means of instructional support and to what extent transactive argumentation mediates their effects on students' knowledge about argumentation. We present process measures from a 2x2-factorial experiment with the factors CSCL script and heuristic worked examples conducted with *N*=101 prospective math teacher students. Results show that both means of instructional support induced transactive argumentation in the collaborative learning process. The self-generated transactive argumentation, but not the partner-generated transactive argumentation mediated the effects of both types of instructional support on students' development of argumentation knowledge. Nevertheless, the learning partners mutually influenced their transactive argumentation.

Introduction

Over the last decade CSCL research has focused on argumentation as a goal of educational interventions. To foster students' argumentation and to support them to develop the corresponding skills, various instructional approaches (e.g. CSCL scripts, representational guidance) have been designed and evaluated across various domains (for an overview, see Noroozi, Weinberger, Biemans, Mulder, & Chizari, 2012). Argumentation skills are also important in mathematical discourse, particularly for working on mathematical proof problems. During the process of mathematical proof, argumentation skills are required at different points (Aberdein, 2009). Yet, students often have problems to construct arguments in general as well as in mathematical context. For example, Sadler (2004) summarized that students show serious difficulties in socioscientific argumentation (e.g.: they do not justify claims, they do not take any counter-argument into account, etc.). Within the mathematical domain, Heinze, Reiss, and Rudolph (2005) found that high school students were able to solve problems, requiring one single argument, but failed in producing logical chains of more than one argument in mathematical argumentation tasks. Thus, students' efforts to acquire argumentation skills within the mathematical context should be supported by using adequate instruction. The study presented in this paper is embedded in the context of a project that investigated the effectiveness of two kinds of instructional support on students' acquisition of argumentation skills in mathematical proof tasks. More specifically, a CSCL script adapted from Stegmann, Weinberger, and Fischer (2007) and heuristic worked examples (Reiss & Renkl, 2002) were applied. In previous analyses within this project, we showed that providing students with the CSCL script and heuristic worked examples both had positive effects on students' acquisition of argumentation skills (Kollar, et al., 2012). Yet, it is still not clear which collaborative learning processes led to these effects. Thus, the main purpose of this paper is to provide an analysis of the collaborative learning processes of the dyads in a mathematics learning environment. Especially, we investigate to what extent the two treatments caused transactive argumentation (i.e. learning partners mutually refer to each other using argumentative moves like criticizing). Further we explore if the induced transactive argumentation can explain the effectiveness of both scaffolds on students' development of knowledge about argumentation as a part of argumentation skills. We also analyse if there is a difference of the effectiveness between transactive argumentation the learners generated themselves and transactive argumentation that was generated by their respective learning partner and to what extent the frequency of transactive argumentation expressed by the learning partners mutually influenced each other.

CSCL scripts and heuristic worked examples for mathematical argumentation

Argumentation skills (i.e. the skills to engage in a social-discursive argumentative dialog in an effectual way) might intuitively rather be required in domains like politics or philosophy. Nevertheless, they are also necessary in mathematics (e.g. Schwarz & Linchevski, 2007), in particular when it comes to mathematical proof problems. According to Boero (1999), the proof process consists of the following steps: (1) generation of a conjecture, (2) formulation of a mathematical statement, (3) exploration of the mathematical statement, (4) selection of adequate theorems to generate a proof draft, (5) construction of the proof draft, (6) formulation of the formal proof. At several points within this process, argumentation needs to be applied, e.g., when one has to find or evaluate a conjecture, to chose applicable arguments, or when a formal proof must be presented to a broader public (Aberdein, 2009). Thus, for argumentation in the mathematical domain formal patterns of the construction of single arguments and the social process of argumentation between dialog partners can be found, similar to other domains. For instance, Toulmin's (1958) argument schema is widely used for the evaluation of single arguments (van Eemeren & Grootendorst, 2004). Also, dialectical forms of argumentation - simplified as the cycle of 'argument', 'counter-argument', and 'synthesis' - might function as a common ground for social discursive activities where two or more dialogue partner are engaged in an argumentative discourse (Leitão, 2000). Recently, various kinds of CSCL instructions that support students' acquisition of argumentation skills have been investigated (Noroozi, et al., 2012). One instructional approach that has shown positive effects on the acquisition of argumentation skills is scripting (e.g. Stegmann, et al., 2007). In general, CSCL scripts distribute roles and activities among the learners and sequence activities and role changes to guide students through a collaborative learning process that is beneficial for their learning (King, 2007; Kollar, Fischer, & Hesse, 2006), both with respect to domain-specific knowledge and the internalization of the domain-general skills a script has learners to practice during learning (Fischer, Kollar, Stegmann, & Wecker, 2013). CSCL scripts that are designed for argumentation guide students through argumentative discourses by prompting them to fulfil adequate activities within each step of an argumentative discourse cycle (e.g. Hron, Hesse, Cress, & Giovis, 2000; Weinberger, Stegmann, & Fischer, 2010) or by distributing discussion roles among the learning partners (e.g. De Wever, Van Keer, Schellens, & Valcke, 2010). Studies about CSCL scripts for argumentation have shown positive effects on students' acquisition of domain-general argumentation skills. For instance, the study by Stegmann et al. (2007) could show that students learning with a CSCL script that was based upon the dialectical cycle of argument, counterargument and synthesis (Leitão, 2000) and Toulmin's argument schema (1958) outperformed students learning without collaboration support in developing argumentation skills. But there has not been systematic research on CSCL scripts for argumentation in the mathematical domain.

Yet, scaffolding collaborative learning processes may not be enough to help students acquire argumentation skills. A review by Vogel, Kollar, and Fischer (2012) revealed that the effectiveness of CSCL scripts on the acquisition of domain-specific knowledge can be advanced by combining them with additional instructional support that provides domain-specific content knowledge (e.g. content schema; Ertl, Kopp & Mandl, 2006). An improvement of the effectiveness of CSCL scripts through the simultaneous provision of domain-specific support may also be expected for the acquisition of domain-general skills (e.g. argumentation skills), since domain-specific instructional support provides content knowledge that can be more deeply elaborated when collaboration is guided by a script. When two kinds of instructional support are used in combination, at least an additive effect would be desirable, i.e. that the (positive) effects of two kinds of instructional support add up when applied together, but do not positively amplify each other. The optimum for the combination of two kinds of instructional support would be synergistic scaffolding (Tabak, 2004). Given the expectation of achieving synergistic scaffolding when combining a CSCL script with a domain-specific instructional support, *heuristic worked examples* were implemented as domain-specific instructional support for the present study. Generally, worked examples provide students with an elaborated worked out solution that is exemplary for solving the type of problem tasks assigned to the learners. While traditional worked examples (e.g. Atkinson, Derry, Renkl, & Wortham, 2000) have shown to be helpful for the acquisition of skills needed to solve rather well-defined problems, they lack of a flexible access to the heuristics strategies that underlie the process of solving rather complex problems (e.g. mathematical proof problems). To adapt the traditional worked examples to the needs of solving complex mathematical proof problems, heuristic worked examples have been developed (Reiss & Renkl, 2002) that describe an authentic solution process according to a process model, e.g. Boero's (1999) experts' model, and provide heuristic strategies. A study by Hilbert, Renkl, Kessler, and Reiss (2007) showed a positive effect of learning with heuristic worked examples compared to regular instruction on teacher students' geometry concepts and proof skills. For the study presented in this paper Boero's process model served as basis for the development of the used heuristic worked examples.

Transactive argumentation within the collaborative learning process

There are many types of discourse activities that contribute to individual learning. Chi (2009) differentiates between active, constructive and interactive learning processes. Active processes are observed when something is physically done with information by the learner. Constructive processes are characterized as the production of knowledge beyond the information the learner decodes from the learning material. Finally, interactive processes are characterized as collaborative processes in which the learners take each partner's contribution into account. When learning collaboratively, all three types of learning processes are possible to occur. However, what according to Chi (2009) really makes collaborative learning effective, are interactive processes. Others have called such processes "transactive" (Teasley, 1997; Weinberger & Fischer, 2006). Given the high potential ascribed to the interactive resp. transactive learning processes, in this paper we use the transactivity principle for learning with CSCL scripts that was stated in the Script Theory of Guidance for Computer-Supported Collaborative Learning (Fischer, et al., 2013). According to this principle, CSCL scripts will be more beneficial for learning the more they induce a transactive learning process, i.e. the more they lead to the learning partners' mutually referring to each others contributions. Recent studies have shown the impact transactive CSCL scripts can have on learning (e.g. Noroozi, Teasley, Biemans, Weinberger, & Mulder, in press). In this study we specifically focus on transactivity of argumentation, i.e. the extent to which learners refer to their partners' contributions in an argumentative way, for example through criticizing or synthesizing the partners' arguments. Through actively building on each other's arguments, optimally a deep elaboration of both the argument and the underlying concepts is achieved. Thus, transactive argumentation should also lead to higher learning success. Furthermore, the repeated use of transactive argumentation during the collaborative learning process should lead to the internalization of a script, when it is designed to help students engage in transactive argumentation, which is the case for the CSCL script used in this study. One open issue is whether an individual's learning is dependent on his/her self-generated transactive argumentation, or whether it is (also) dependent on partner-generated transactive argumentation. It may be argued that, for the acquisition of argumentation skills it is more important for the learner to construct transactive argumentation her-/himself than to be exposed to the learning partner's transactive argumentation, because to actively generate transactive argumentation the learner must deal intensively with the partner's contribution, whereas the learner might not necessarily process the partner-generated transactive argumentation at all. Nevertheless, the learning partners might mutually influence each others' contributions by the transactive argumentation they express.

Research Questions

The research questions this paper tries to answer are:

(RQ1) What are the effects of a CSCL script, heuristic worked examples and the combination of both on students' use of transactive argumentation when collaboratively working on mathematical proof tasks? We expected a positive effect of the CSCL script compared to unscripted collaborative learning on the use of transactive argumentation, since CSCL scripts sequence the learners' collaborative learning activities by inducing advantageous activities (e.g. referring to the learning partner's contribution) at the appropriate point of time during the collaboration process. The heuristic worked examples provide domain-specific content that facilitates learners to construct arguments, especially when the construction of arguments is supported by the CSCL script (Sadler, 2004). Thus, for the heuristic worked examples compared to learning without heuristic worked examples we expected a positive effect as well, and for the combination of both scaffolds we expected to find a synergistic scaffolding effect (Tabak, 2004).

(RQ2a) To what extent are the effects of a CSCL script and heuristic worked examples on knowledge about argumentation mediated by self-generated transactive argumentation? We expected that self-generated transactive argumentation explains (i.e. mediates) the positive effect of the CSCL script on the acquisition of knowledge about argumentation. When learners carry out what a CSCL script suggests, this should lead to a more frequent use of transactive argumentation than in unscripted discussions and – mediated by transactive argumentation – to an internalization of knowledge about argumentation embedded in the script (Fischer, et al., 2013). Also, a mediation of the effect of learning with heuristic worked examples on the acquisition of knowledge about argumentation through self-generated transactive argumentation was expected. The heuristic worked examples provide domain-specific content students could use to repeatedly engage in an argumentative discourse. Again, this is expected to lead to a more frequent use of transactive argumentation and thus to an internalization of the script for argumentation.

(RQ2b) To what extent are the effects of a CSCL script and heuristic worked examples on knowledge about argumentation mediated by partner-generated transactive argumentation? The effects of both scaffolds on the acquisition of knowledge about argumentation might be mediated by the partner-generated transactive argumentation but not to the same extent as they are expected to be mediated by the self-generated transactive argumentation because the learners do not necessarily have to process the learning partner's contribution.

(RQ3) To what extent do the frequencies of transactive argumentation generated by each of the two learning partners reciprocally influence each other? By definition, the generation of transactive argumentation depends on the contributions the learning partner provides within the collaborative learning process. Therefore, we expected a positive relationship between self- and partner-generated transactive argumentation. Further we expected that a positive effect of the partner-generated transactive argumentation on one's own knowledge about argumentation would be mediated by the self-generated transactive argumentation.

Methodology

Participants and design

The study was conducted as part of a two weeks course for prospective math teacher students that were about to start their university education. Out of 162 students participating in the pre-test, 61 students missed more than one treatment and/or did not show up at the posttest and thus had to be excluded for the purposes of this paper. After clearing for drop-outs, N = 101 math teacher students were included in the analyses presented in this paper. A 2x2-factorial experiment with the independent variables CSCL script (with vs. without) and heuristic worked examples (with vs. without) was established. Participants were randomly assigned to one of the four experimental conditions. The study took place on five consecutive days with pre- and post-test data collection on the first and fifth day. On the second through the fourth day the participants were exposed to one treatment session per day lasting 45 minutes. For each of the three treatment sessions, the learners were randomly assigned to new dyads to reduce the effect one specific participant might have on his or her learning partner.

Setting and learning environment

Students learned collaboratively in dyads in a CSCL environment (see Figure 1) on three different mathematical proof tasks (e.g.: "Take an uneven amount of consecutive numbers and add them up. Repeat this and try to find regularities. Formulate a conjecture and prove it."). The learning partners were each equipped with one laptop and a graphic tablet and worked co-presently on the proof tasks. The laptops of both learning partners were linked to each other to distribute different interconnected prompts and material as well as to display a mirrored workspace where the learners could share their written communication and drawings (see Figure 1). Since the learning partners were allocated face-to-face they were able to speak to each other but they were requested to write their discussion about and progress on the mathematical proof task into the shared work space.

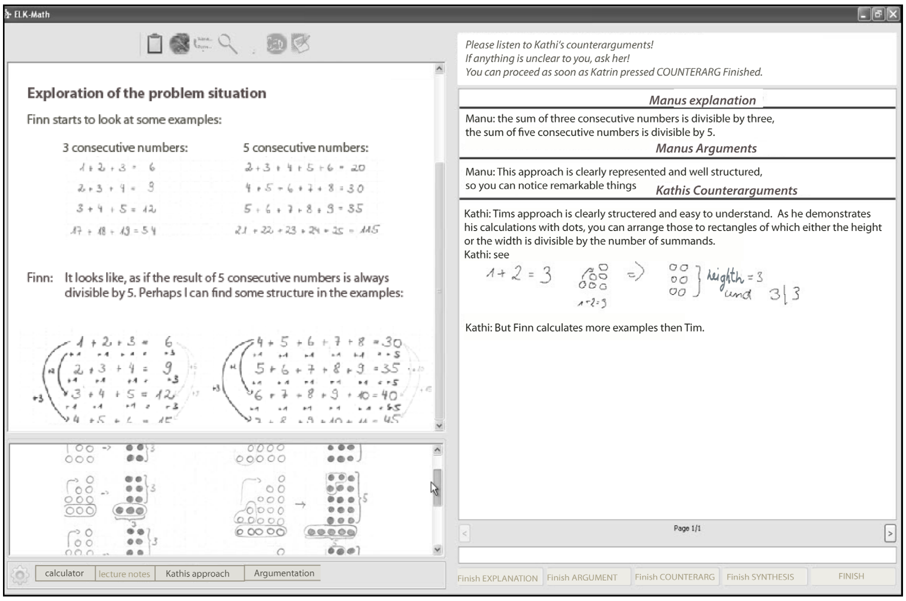

Figure 1. Screenshot of the computer program (left side of the screen: private work space including the problem to be solved resp. the heuristic worked example; right side: shared work space displaying script prompts or not).

The learning environment on the laptop screen was divided into two parts. On the left half of the screen, the learning environment provided the mathematical proof task, a calculator and domain-specific lecture notes (available in all conditions) as well as the heuristic worked example (in the conditions with heuristic worked examples only). On the right half of the screen, the students were able to share text and drawings by using the available text and graphic chat function (available in all conditions). The students had the opportunity to create any number of pages for their written communication and browse through them within the current treatment session. On the upper right side of the screen the script prompts were displayed that aimed at sequencing students' contribution types to the discussion (in the conditions with CSCL script only).

Independent variables

In all experimental conditions, students were requested to work on the proof task alternately individually and collaboratively by discussing their ideas. In the conditions *with CSCL script*, we adapted the script that was investigated by Stegmann et al. (2007) to the context of mathematical argumentation tasks. Thus, the collaborative discussion was sequenced into the three phases (1) argument, (2) counterargument, and (3) synthesis. When prompted to construct counterarguments or synthesize arguments, students were specifically asked to refer to their learning partner's contribution (Leitão, 2000). Also, students were encouraged to formulate sound arguments according to Toulmin's (1958) model of argument construction (including claims, data and rebuttals). In the conditions *without CSCL script,* students discussed their ideas without receiving any guidance for their discussion.

In the conditions *with heuristic worked examples,* students received worked examples that split a possible solution of one proof task into steps of mathematical proof (adapted from Boero, 1999). The worked examples contained different heuristic strategies which an imaginary student applied to make progress within each of these steps. At each first, third and fourth step within the heuristic worked examples (i.e. at (1) generation of a conjecture, (3) exploration of the mathematical statement, (4) selection of adequate arguments to generate a proof draft), both students were provided with different example versions. After studying these specific steps individually, students were asked to present both versions of the step to each other and to discuss them. In the conditions *without heuristic worked examples*, students had to work on the mathematical proof tasks by problem solving only, i.e. without receiving guidance neither on the steps of mathematical proof nor on heuristic strategies. The students in these conditions were alternately asked to think about their ideas how to solve the problem individually and to present their ideas to each other and discuss them.

Dependent variables

For the pre- and post-test measure of *knowledge about argumentation*, students were requested to describe typical phases and activities they would expect to occur in a discussion about a science topic. A topic different from mathematics was chosen to investigate knowledge about argumentation that could be transferable to different domains. Students answers were coded for the amount of argumentative elements (e.g. pro-argumentation, counter-argumentation, etc.) they named correctly (for further information, e.g., on reliabilities, see Kollar, et al., 2012).

To measure *the frequency of transactive argumentation,* the written communication of the learning partners during the three treatment sessions were coded. As transactive argumentation, all statements were counted that built on partner's contribution in an argumentative way. To code learners' written communication, first the pages created in each of the three treatment sessions were segmented at the points where turn taking occurred. The resulting segments were taken as unit of analysis to code the written communication by coders that were trained as follows. The research assistant responsible for the study trained two student assistants to code the segments regarding transactive argumentation using a coding scheme with descriptions and examples of transactive argumentation (see Table 1). The training was conducted within eight weeks with alternately coding of the training material and discussing coding differences together with the research assistant to reach a more precise coding. The frequencies of segments containing transactive argumentation were then summed up for each learner separately. During the training the inter-rater reliability could be advanced from poor values (ICC_{unjust} < .40) to sufficient values (ICC_{unjust} > .60). After training, the two coders coded a sample of > 5% of the whole sample of written communication across all conditions and treatment sessions with sufficient inter-rater reliability (ICC_{unjust} = .68).

Table 1: Excerpt from the coding scheme with descriptions and examples for transactive argumentation.

Transactive argumentation Description	Examples
Criticizing: Comments that tackle the approach to solve the problem or the solution itself and contain counterargumentation and/or criticism directly referring to the learning partner's contribution	- "… but your description of the problem space is less helpful because not every kind of possible solutions can be displayed" - "…2 + 3 = 5" (as counter-example to the claim that the sum of two consecutive numbers is always even)
Synthesizing: Comments that synthesize previous contributions with containing at least one contribution made by the learning partner	- "…the summary of the pros and cons we made is…" - "Taking your criticism into account we could agree on distinguishing between cases when the numbers are even and uneven"

Statistical analyses

To answer our research questions, we used univariate analysis of variance (to test the effects of the two treatments on the frequency of transactive argumentation) and linear regressions (to determine to what extent self- and partner-generated transactive argumentation would be a predictor for the knowledge about argumentation displayed in the posttest). To confirm the significance of mediating predictors in the linear regression models, we calculated Sobel tests (Sobel, 1982). For all tests the significance level was set to α = .05. As measures of effect sizes, partial η^2 were used, with values between .01 and .05 being considered as weak effects, values between .06 and .14 as medium effects, and values of .14 and higher as large effects (Cohen, 1988).

Results

(RQ1) An ANOVA revealed a positive effect of learning with the CSCL script on the frequency of transactive argumentation during the collaboration process compared to learning without the CSCL script ($F(1,97) = 11.63$, $p = .001$, partial $\eta^2 = .11$). Also the effect of learning with heuristic worked examples on the frequency of transactive argumentation compared to learning without heuristic worked examples was positive ($F(1,97) = 28.41$, $p < .001$, partial $\eta^2 = .23$). The interaction effect was significant, ($F(1,97) = 18.24$, $p < .001$, partial $\eta^2 = .16$). Post-hoc comparisons of the four experimental groups showed that only the learners supported with both forms of instructional support at once achieved significantly higher frequencies of transactive argumentation than learners in the other three groups ($F(1,97) = 57.77$, $p < .001$, partial $\eta^2 = .37$).

(RQ2a) Learning with the CSCL script and learning with the heuristic worked examples positively predicted the acquisition of knowledge about argumentation measured between pre- and post-test (model 1; see Table 2 for exact β-values in the linear regression models; see also Kollar, et al., 2012). Both positive predictions disappeared when the frequency of self-generated transactive argumentation was integrated into the regression model (model 2a), while the frequency of self-generated transactive argumentation predicted significantly the acquisition of knowledge about argumentation. Sobel tests showed that self-generated transactive argumentation significantly mediated the effect of the CSCL script ($z = 2.26$, $p = .01$, one-tailed) and the effect of the heuristic worked examples ($z = 2.32$, $p = .01$, one-tailed) on students' acquisition of knowledge about argumentation.

(RQ2b) When the frequency of partner-generated transactive argumentation was included into the initial linear regression model, it did not serve as a significant predictor for students' development of knowledge about argumentation, and the CSCL script still positively predicted students' development of knowledge about argumentation significantly (model 2b). The Sobel test could also not confirm partner-generated transactive argumentation as mediator for the effect of the CSCL script on the acquisition of knowledge about argumentation ($z < 1$, ns). In contrast, heuristic worked example were no longer a significant predictor of knowledge about argumentation when partner-generated transactive argumentation was included into the initial linear regression model (model 2b). However, Sobel tests did not confirm that the effect of heuristic worked examples on knowledge about argumentation was mediated by the partner-generated transactive argumentation as the reduction of the β-value of heuristic worked examples between the model was not substantial ($z < 1$, ns).

Table 2: Summary of multiple regression models with predictors for knowledge about argumentation.

Variable	B	SE B	β
Model 1			
CSCL Script (C)	0.902	0.355	.332**
Heuristic Worked Examples (H)	0.664	0.375	.245*
C X H	-0.258	0.516	-.083
Model 2a			
CSCL Script (C)	0.416	0.398	.153
Heuristic Worked Examples (H)	0.058	0.440	.021
C X H	0.281	0.549	.091
Self-generated transactive argumentation per treatment session	0.549	0.222	.290**
Model 2b			
CSCL Script (C)	0.845	0.400	.311*
Heuristic Worked Examples (H)	0.598	0.432	.220
C X H	-0.199	0.553	-.064
Partner-generated transactive argumentation per treatment session	0.055	0.177	.036

*< .05, **< .01, one-tailed

(RQ3) For the analysis of the extent learners of a dyad might have mutually influenced each other in their generation of transactive argumentation, a linear regression model revealed a significant positive relationship between the frequencies of the partner-generated and the self-generated transactive argumentation (stand. $\beta = .655$, $p < .001$). Further, the frequency of partner-generated transactive argumentation positively predicted the acquisition of knowledge about argumentation measured between pre- and post-test (model 3, see table 3 for exact β-values). The positive prediction for partner-generated transactive argumentation disappeared when the frequency of self-generated transactive argumentation was integrated into the regression model (model 4), while the frequency of self-generated transactive argumentation predicted significantly the acquisition of knowledge about argumentation. Sobel tests showed that self-generated transactive argumentation significantly mediated the positive effect of the frequency of partner-generated transactive argumentation ($z = 3.12$, $p = .002$, one-tailed).

Table 3: Summary of multiple regression models with predictors for knowledge about argumentation.

Variable	B	SE B	β
Model 3			
Partner-generated transactive argumentation per treatment session	0.304	0.152	.198*
Model 4			
Partner-generated transactive argumentation per treatment session	-0.113	0.191	-.073
Self-generated transactive argumentation per treatment session	0.783	0.235	.414**

*< .05, **< .01, one-tailed

Conclusion and Discussion

The interaction effect between the CSCL script and the heuristic worked example indicates that both types of instructional support applied together produce synergistic effects (Tabak, 2004) on the use of transactive argumentation within the collaborative learning processes. Thus, the CSCL script and the heuristic worked examples amplified each other's effects on the collaborative process i.e. the effectiveness of the CSCL script was increased by the heuristic worked examples. Thus, when students were supported with heuristic worked examples, the CSCL script could induce transactive argumentation more effectively by guiding students through a sequence of argumentative discourse moves containing arguments, counterarguments and syntheses. This might have been caused by the richer content the heuristic worked examples provided on which students were better able to apply the prompts of the CSCL script that specifically aimed for a high transactivity during argumentation (Sadler, 2004). Further, the results underpin the importance of transactivity in collaborative learning processes (Fischer et al., 2013, Teasley, 1997) for individual skill acquisition. The self-generated transactive argumentation induced by the script and the heuristic worked examples significantly mediated the positive effects of both means of instructional support on students' advances in their knowledge about argumentation. Thus, it can be recommended to carefully design instructional interventions to foster students' knowledge about argumentation by focussing on ways that are likely to induce transactive argumentation. Interestingly, only self-generated transactive argumentation, but not the transactive argumentation generated by the learning partner was influential for the acquisition of knowledge about argumentation. Thus, for the acquisition of one's own knowledge about argumentation it might be more important that learners generate transactive argumentation by themselves than to be exposed to a learning partner who is generating transactive argumentation (Teasley, 1997). This makes sense, as learners have to be engaged in the partner's contribution when generating transactive argumentation while it is not necessary for them to process as deeply with the transactive argumentation which is generated by their learning partners. Nevertheless, the learning partner is still important for a beneficial learning process, since transactive argumentation needs the learning partner's contributions to refer to them. As the regression analyses with respect to RQ3 show, both learning partners mutually influenced their generation of transactive argumentation within the learning process. The importance of the learning partner is supported by results showing that both self-generated and partner-generated transactive argumentations are substantially positively related. Further, an indirect effect of partner-generated transactive argumentation on one's own development of knowledge about argumentation was found to be fully mediated by self-generated transactive argumentation. This means that it is not enough to just be exposed to transactive arguments but these arguments have to be transactively processed by the learners. Thus, as a further theoretical conclusion, the transactivity principle in the script theory of guidance (Fischer et al., 2013) has to be differentiated for the mathematical context that was used in this study. For the design of CSCL scripts to foster argumentation skills in mathematical context, it can be suggested that it should aim to induce transactive argumentation for each of the learning partners.

References

Aberdein, A. (2009). Mathematics and argumentation. *Foundation of Science, 14*(1-2), 1-8.

Atkinson, R.K., Derry, S.J., Renkl, A. & Wortham, D. (2000). Learning from Examples: Instructional Principles from the Worked Examples Research. *Review of Educational Research, 70*(2), 181-214.

Boero, P. (1999). Argumentation and mathematical proof: A complex, productive, unavoidable relationship in mathematics and mathematics education. *International Newsletter on the Teaching and Learning of Mathematical Proof, July/August 1999.* http://www.lettredelapreuve.it/.

Chi, M. T. H. (2009). Active-Constructive-Interactive: A Conceptual Framework for Differentiating Learning Activities. *Topics in Cognitive Science, 1*(1), 73–105.

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). Hillsdale, NJ: Erlbaum.

De Wever, B., van Keer, H., Schellens, T. & Valcke, M. (2010). Roles as a structuring tool in online discussion groups: The differential impact of different roles on social knowledge construction. *Computers in Human Behavior, 26*(4), 516-523.

Ertl, B., Kopp, B., & Mandl, H. (2006). Fostering collaborative knowledge construction in case-based learning in videoconferencing. *Journal of Educational Computing Research*, *35*(4), 377–397.

Fischer, F., Kollar, I., Stegmann, K. & Wecker, C. (2013). Toward a Script Theory of Guidance in Computer-Supported Collaborative Learning. *Educational Psychologist, 58*(1), 56-66.

Heinze, A., Reiss, K., & Rudolph, F. (2005). Mathematics achievement and interest from a differential perspective. *Zentralblatt für Didaktik der Mathematik, 37(3),* 212-220.

Hilbert, T. S., Renkl, A., Kessler, S., & Reiss, K. (2008). Learning to prove in geometry: Learning from heuristic examples and how it can be supported. Learning and Instruction, 18(1), 54–65.

Hron, A., Hesse, F. W., Cress, U., & Giovis, C. (2000). Implicit and explicit dialogue structuring in virtual learning groups. *British Journal of Educational Psychology*, *70*, 53–64.

King, A. (2007). Scripting collaborative learning processes: A cognitive perspective. In F. Fischer, I. Kollar, H. Mandl & J. M. Haake (Eds.), *Scripting computer-supported collaborative learning: Cognitive, computational, and educational perspectives* (pp. 13-37). New York: Springer.

Kollar, I., Fischer, F., & Slotta, J. D. (2007). Internal and external scripts in computer-supported collaborative inquiry learning. *Learning and Instruction, 17*(6), 708–721.

Kollar, I., Ufer, S., Lorenz, E., Vogel, F., Reiss, K., & Fischer, F. (2012). Using heuristic worked examples and collaboration scripts to help learners acquire mathematical argumentation skills. In J. van Aalst, K. Thompson, M. J. Jacobson, & P. Reimann (Eds.), *The future of learning: Proceedings of the 10th international conference of the learning sciences (ICLS 2012) – Volume 1, Full Papers* (pp. 331–338). Sydney, Australia: ISLS.

Leitão, S. (2000). The potential of argument in knowledge building. *Human Development, 43*, 332–360.

Noroozi, O., Teasley, S. D., Biemans, H. J. A., Weinberger, A., & Mulder, M. (in press). Facilitating learning in multidisciplinary groups with transactive CSCL scripts. *International Journal of Computer-Supported Collaborative Learning.*

Noroozi, O., Weinberger, A., Biemans, H. J.A., Mulder, M., & Chizari, M., (2012). Argumentation-Based Computer Supported Collaborative Learning (ABCSCL): A synthesis of 15 years of research. *Educational Research Review, 7*(2), 79-106.

Reiss, K. & Renkl, A. (2002). Learning to prove: The idea of heuristic examples. *Zentralblatt für Didaktik der Mathematik, 34*(1), 29-35.

Sadler, T. D. (2004). Informal reasoning regarding socioscientific issues: A critical review of research. *Journal of Research in Science Teaching*, *41*(5), 513–536.

Schwarz, B. B., & Linchevski, L. (2007). The role of task design and argumentation in cognitive development during peer interaction: The case of proportional reasoning. *Learning and Instruction*, *17*(5), 510–531.

Sobel, M. E. (1982). Asymptotic confidence intervals for indirect effects in structural equation models. *Sociological Methodology*, *13*, 290–312.

Stegmann, K., Weinberger, A., & Fischer, F. (2007). Facilitating argumentative knowledge construction with computer-supported collaboration scripts. *International Journal of Computer-Supported Collaborative Learning, 2*(4), 421-447.

Tabak, I. (2004). Synergy: A Complement to Emerging Patterns of Distributed Scaffolding. *The Journal of the Learning Sciences, 13*(3), 305–335.

Teasley, S. D. (1997). Talking about reasoning: How important is the peer in peer collaborations? In C. O'Malley (Ed.), *Discourse, Tools, and Reasoning: Situated Cognition and Technologically Supported Environments* (pp. 361–384). Berlin: Springer.

Toulmin, S. E. (1958). *The uses of argument*. Cambridge: Cambridge University Press.

Van Eemeren, F. H. & Grootendorst, R. (2004). *A Systematic Theory of Argumentation: The pragma-dialectical approach.* Cambridge: Cambridge University Press.

Vogel, F., Kollar, I., & Fischer, F. (2012). Effects of computer-supported collaboration scripts on domain-specific and domain-general learning outcomes: A meta-analysis. In J. van Aalst, K. Thompson, M. J. Jacobson, & P. Reimann (Eds.), *The future of learning: Proceedings of the 10th international conference of the learning sciences (ICLS 2012) – Volume 2, short papers, symposia, and abstracts* (pp. 446–450). Sydney, Australia: ISLS.

Weinberger, A., & Fischer, F. (2006). A Framework to analyze argumentative knowledge construction in computer-supported collaborative learning. *Computers and Education, 46*(1), 71–95.

Weinberger, A., Stegmann, K., & Fischer, F. (2010). Learning to argue online: Scripted groups surpass individuals (unscripted groups do not). *Computers in Human Behavior. 26*, 506-515.

Acknowledgments
The research reported here was funded by the Deutsche Forschungsgemeinschaft (DFG).