Conflicts and Collaboration: A Study of Upper Elementary Students Solving Computer Science Problems

Jennifer Tsan, North Carolina State University, jtsan@ncsu.edu
Jessica Vandenberg, North Carolina State University, jvanden2@ncsu.edu
Xiaoting Fu, North Carolina State University, xfu7@ncsu.edu
Jamieka Wilkinson, University of Florida, wilkinsonj2013@ufl.edu
Danielle Boulden, North Carolina State University, dmboulde@ncsu.edu
Kristy Elizabeth Boyer, University of Florida, keboyer@ufl.edu
Collin Lynch, North Carolina State University, cflynch@ncsu.edu
Eric Wiebe, North Carolina State University, wiebe@ncsu.edu

Abstract: Collaborative learning holds great value for young learners. However, these learners often encounter conflicts that may arise based on the learning task itself or on external factors. Little is known about the nature and evolution of conflicts that occur when elementary learners work together. To investigate this phenomenon, we analyzed videos of six pairs of students completing a programming activity. We identified four stages of conflict: *initiation*, *escalation*, *de-escalation*, and *conclusion*. Our analysis showed that the conflicts typically began around disagreements about code, who should have control of the keyboard and mouse, and other interpersonal events. Also, we found that some pairs of students resolved their conflicts through self-explanation and listening while others did not take advantage of those constructive strategies. This research reveals some ways in which conflicts evolve between elementary learners, and how we may be able to support them in conflict resolution while solving problems.

Introduction

Collaborative learning is a complex process that involves co-constructing knowledge and maintaining shared ideas (Roscelle & Teasley, 1995, p. 70). Within computer science (CS), pair programming is a widely used collaborative learning paradigm in which two learners work on one computer, taking turns with the control. The learner controlling the computer is the *driver* and the other, the *navigator*, plans ahead and looks for mistakes. Many empirical studies have shown the benefits of pair programming for college-aged learners, including the promotion of desirable practices such as pre-planning (Porter & Simon, 2013). In light of these benefits, pair programming has recently been used when supporting younger CS learners. However, emerging observations suggest that young learners need additional support to benefit from this collaborative paradigm (Shah, Lewis, & Caires, 2014). Since most research on CSCL with young learners has focused on non-CS topics (e.g., Olsen, Rummel, & Aleven., 2017), collaborative programming with young learners is an emerging area of research.

We have recently begun to implement pair programming with students at the 4th and 5th grade level. Our observations of multiple elementary CS classrooms led us to believe that conflicts between the students are unlike those we had observed in our extensive studies of older CS students (Rodríguez, Price, & Boyer, 2017) and studying conflicts is a crucial open area for research. Therefore, we investigated the following research question: *How do conflicts arise and evolve when elementary learners engage in pair programming?* We analyzed pair programming dialogue of elementary students and categorized video excerpts of the students into stages of conflict based on previous work (Jeong, 2008; Rubin, Pruitt, & Kim, 1994), which may not all be present in each conflict: initiation, escalation, de-escalation, and conclusion. In addition, some conflicts had a resolution. Our findings highlight the importance of better understanding how elementary students collaborate while learning CS.

Study context and participants

We collected data from a pair programming study conducted at two U.S. elementary schools. A total of 40 4th and 5th grade students (typically 8 to 11 years old) participated and provided parental consent and student assent, 20 in each school. We collected webcam video and screen capture data from each pair of students over a period of 5 weeks. The curriculum was designed by the authors and covered CS topics appropriate for 4th and 5th grade students using a block-based programming language where users write code by dragging, dropping, and connecting programming blocks.

Results and discussion

Every conflict episode had a clear start and a conclusion but not all of them moved through a full cycle of escalation, de-escalation, or resolution. We identified 38 total conflict episodes (simply 'conflicts' for short)

from six programming videos in our analysis, with an average of 6.4 conflicts per 45-minute video. We found that the majority of conflicts between the students (71%) went unresolved. Additionally, many pairs engaged in more escalation than de-escalations. This suggests that the increase of intensity in their conflicts were due in part to the learners having difficulty working through their problems. The descriptive statistics of the conflicts in each pair are displayed in Table 1.

Table 1: Information about each pair, including descriptive statistics of the conflicts

Evans	1	7	18s | 6s | 40s	57%	43%	29%
	2	8	18s | 8s | 39s	63%	63%	0%
	3	5	53s | 8s | 3m 4s	40%	40%	40%
Adams	4	4	17s | 4s | 39s	75%	50%	0%
	5	9	35s | 9s | 1m 32s	89%	89%	44%
	6	5	12s | 8s | 22s	20%	20%	60%

While analyzing the conflicts, we noticed that they were often triggered by one of three issues: disagreements about changes in the code, disputes over who should drive, and non-CS problems (e.g. camera position, personality/social disagreements). As the conflicts unfolded, the disagreements sometimes evolved into verbal arguments and physical incidents over control of the equipment, or with each other.

Young learners do not naturally have collaborative strategies (Gillies & Ashman, 1996). This is illustrated by four pairs of learners who argued with each other frequently and often repeated suggestions but did not offer justifications or explanations. In our data, we found cases in which students were focused on the task at hand but encountered difficulty resolving their conflicts. In both cases, the navigator gave a suggestion which was rejected or ignored by the driver and the conflict escalated but did not get resolved. In the cases where pairs appeared to successfully resolve their conflicts, the navigators were engaged and contributed to the process. Through the use of collaborative dialogue strategies such as self-explanation the students resolved their conflicts. This is consistent with prior research that suggested that groups of young learners who accept suggestions or enter into discussions about them collaborate more successfully than those that reject suggestions (Barron, 2003).

Conclusion

While collaboration has been shown to be beneficial in many situations, including CS learning, supporting younger learners in collaborative CS learning brings new challenges that we must be cognizant of, including the students' level of socio-emotional development. The results described in this paper illustrated three patterns of collaborative behavior: conflicts arising as a result of distractions, conflicts arising due to a lack of collaborative strategies, and conflicts being resolved using collaborative strategies, including listening and self-explanation. Since a lack of resolution may indicate a lack of positive change in the learners' relationships, researchers should develop interventions to support learners in resolving conflicts and developing overall healthier working relationships. Our video analysis points to the need for careful design and study of collaborative scaffolding for students of this age. Future work should continue to investigate aspects of collaboration in CS, including conflicts and how to teach collaboration skills to elementary students. This work can support future research on the design of adaptive support mechanisms to mitigate conflicts between younger learners during collaborative problem-solving, or to prevent them before they arise.

References

Barron, B. (2003). When Smart Groups Fail. *The Journal of the Learning Sciences*, *12*(3), 307-359

Gillies, R. M., & Ashman, A. F. (1996). Teaching Collaborative Skills to Primary School Children in Classroom-Based Work Groups. *Learning and Instruction*, *6*(3), 187-200.

Jeong, H. W. (2008). *Understanding Conflict and Conflict Analysis*. Sage.

Olsen, J. K., Rummel, N., & Aleven, V. (2017). Learning Alone or Together? A Combination can be Best!. *Proceedings of the 12th Intl Conference on Computer Supported Collaborative Learning*, 95-102.

Porter, L., & Simon, B. (2013). Retaining Nearly One-Third More Majors with a Trio of Instructional Best Practices in CS1. *Proceedings of the 44th SIGCSE*, 165-170.

Rodríguez, F. J., Price, K. M., & Boyer, K. E. (2017). Expressing and addressing Uncertainty: A study of collaborative Problem-solving dialogues. *Proceedings of the 12th International Conference on Computer Supported Collaborative Learning*, 207-214.

Rubin, J. Z., Pruitt, D. G., & Kim, S. H. (1994). *Social conflict: Escalation, stalemate, and settlement*. Mcgraw-Hill Book Company.

Shah, N., Lewis, C., & Caires, R. (2014). Analyzing Equity in Collaborative Learning Situations: A Comparative Case Study in Elementary Computer Science. *Proceedings of the 10th ICLS*, 495-502.