Supporting Guided Inquiry with Cooperative Learning in Computer Organization

Yeajin Ham  
Dept. of Educational Psychology & Learning Sciences  
University of Iowa  
Iowa City, Iowa, USA  
yeajin-ham@uiowa.edu

Brandon Myers  
Dept. of Computer Science  
University of Iowa  
Iowa City, Iowa, USA  
brandon-d-myers@uiowa.edu

ABSTRACT
The computer organization course must help students acquire difficult conceptual knowledge and design skills, and improve their teamwork skills for subsequent project courses. Prior research supports that cooperative learning, in which students work together to achieve common goals, may address these challenges. We studied whether increasing the amount of guided inquiry activities and the cooperative support for it (teams and reflection) in an intermediate computer science course would improve achievement and engagement. The intervention group had lower scores on one of two achievement measures, lower engagement, and lower task value of collaborative activities. Qualitative analysis showed that students valued hands-on learning yet resisted guided inquiry, suggesting that sharing the purpose of each type of activity is important. Furthermore, the results showed that students valued learning with peers but were frustrated by group dysfunction, suggesting that instructors must address teamwork comprehensively to realize the benefits of cooperative learning.

KEYWORDS
guided inquiry; cooperative learning; collaborative learning; computer organization; computer architecture; POGIL

1 INTRODUCTION
The computer organization course presents two instructional challenges. First, students struggle to acquire both conceptual knowledge [1], [2] and design skills. Second, at the University of Iowa, this prerequisite to upper-level coursework must prepare students for team projects.

A pedagogy called Peer Instruction [3] has shown promise for increasing achievement in computer science courses over traditional instruction [4]. These achievement gains rely on peer interaction [5][6], but students do not work in teams. One type of interaction that improves both achievement and teamwork skills is cooperative learning [7], in which students recognize the need to work together to achieve common goals.

We studied whether increased use and support of cooperative learning improved achievement, as well as constructs that mediate achievement, in two sections of a computer organization course. In the control section, students spent half the sessions on active learning lectures, lectures interspersed with interactive activities, and half in informal groups working on lab assignments and guided inquiry. In the intervention section, we used additional supports for cooperative learning [9]. First, to support positive interdependence, that students share goals and recognize that they depend on each other to achieve them, students worked in stable teams. Second, to support promotive interaction, that the activities are worthy of group work, teams worked solely on lab assignments and guided inquiry. Finally, to support group processing, that students reflect on how to improve their team, teams completed reflections on teamwork and communication.

We measured achievement using scores on identical exam questions and a concept inventory. To explain achievement results, we measured task value, engagement, and perceptions of peer academic support. From open-ended post-test responses, we found themes for why students in each section found certain tasks helpful to their learning. Our contributions apply to intermediate level computer science courses taught with active learning:

• A quasi-experimental test of increased use of structures supporting cooperative learning
• Insights into why students view activities as helpful and unhelpful to their learning

The quantitative results of our study were null or negative. Students in the intervention showed lower achievement and engagement. However, the qualitative results of our study support that with comprehensive support for cooperative learning, this type of instruction could be effective.

2 BACKGROUND & RELATED WORK

2.1 Cooperative learning

There is evidence that active learning relies on peer interaction to produce learning gains [5][6][10]. A type of peer interaction called cooperative learning is often characterized by positive interdependence (students share goals and work together as a team to achieve them), individual accountability, promotive interaction (the work is worthy of a group discussion), the appropriate use of collaborative skills, and group processing (reflecting on interactions to improve them) [11][12]. Meta-analyses of cooperative learning in college courses found meaningful improvements in achievement [13]–[15]. Cooperative learning in CS1 was associated with lower rates of failure and withdrawal in CS2 [16] and increased achievement and programming self-efficacy if the instructor was trained in the pedagogy [17].

2.1.1 Supporting positive interdependence. An instructor can give multi-faceted problems, assigning different roles or expertise to group members, and giving bonuses for all individuals scoring well on an assessment [7]. This interdependence may be less present in Peer Instruction, where peers collaborate but are not accountable for each other. In our intervention group, we promoted positive interdependence by keeping students in teams for at least 4 weeks and assigning roles.

2.1.2 Supporting group processing. An instructor can have students reflect on the development of their teamwork and communication skills [18] and set and evaluate goals [7]. Our intervention supported group processing by having teams periodically answer prompts at the end of the session about how well the team is working.

2.1.3 Cooperative learning in computer organization. Titterton et al. [19] described a “lab-centric” computer organization course that included collaborative exercises, but it lacked positive interdependence. Experiments were limited to comparisons with traditional lecture-lab instruction [20]. Bruce described a VLSI course that included teams with designated roles working on circuit design problems [21] but did not have a baseline. Arbelaitz et al. [22] described a Computer Architecture course with cooperative learning to scaffold the teamwork of subsequent project-based learning. For each of three intervention semesters, they found increased achievement over traditional instruction; however, they did not compare achievement between the active learning offerings. The cooperative learning instruction strategy that we adopted in our course was Process-Oriented Guided Inquiry Learning (POGIL).

2.3 POGIL

POGIL [23] is a classroom instruction strategy that combines cooperative learning and guided inquiry, in which students construct then apply new concepts in an experiential learning cycle [24]. In a POGIL learning cycle, students answer exploration questions about a model (e.g., data table, flowchart, code), which leads them to apply reasoning in concept invention (e.g., writing a definition, deriving a mathematical relationship). Finally, students answer application questions to apply the concept.

POGIL incorporates elements of cooperative learning. Constructing and applying new concepts in a guided inquiry activity requires students to discuss and challenge each other’s assumptions and inferences, which is promotive interaction. Students work in groups of four and have roles to support positive interdependence. Furthermore, the activities develop students’ process skills, including cognitive skills like critical thinking and social skills like communication and teamwork.

POGIL has been used in several disciplines including CS. Compared to traditional instruction, POGIL increased performance and engagement in college chemistry courses [25] and improved student achievement in CS1 [4]. The CS-POGIL project [26] has created and shared over 200 inquiry activities for CS, including some for computer organization. We used POGIL during both sections of our quasi-experiment, and the intervention group dedicated more than twice as many class sessions to it. We supported this increase with more emphasis on structures that make cooperative learning effective.

3 METHOD

The research questions for our work were:

RQ1: In computer organization, does replacing active learning lectures with additional guided inquiry and labs supported by teams and daily reflection improve student achievement?

RQ2: In computer organization, does replacing active learning lectures with additional guided inquiry and labs supported by teams and daily reflection improve engagement and factors that mediate engagement: peer academic support and task value?

To answer these questions, we conducted a quasi-experimental study over two semesters. Students in one semester experienced the control and students in the other experienced the intervention.

3.1 Context

We ran the study in the computer organization course during Fall 2017 and Spring 2018 semesters. The instructor taught the course both semesters in the same classroom, days, and time of day. Contact time was 50 minutes 3 days/week in a student-centered classroom with distributed laptops, screens, and whiteboards [27][28].

3.1.2 Preparation. Prior to the study semesters, the instructor taught the course twice using active learning lectures and once using the added approaches—labs and guided inquiry. That third offering was like the control semester. The instructor also had 30
Students took on four common POGIL roles of Manager (keep on track), Recorder (write answers), Speaker (present answers), and Reflector (elicit balanced contributions, fill in exit slip during intervention). Guided inquiry is a good match to the learning objectives of computer organization. Students invent concepts inductively by recognizing patterns (e.g., behavior of a program or circuit) or deductively by evaluating proposed solutions to a problem (e.g., various ways to implement procedure calls in assembly code). Concept invention is followed by application, where students write and debug programs and circuits. The inquiry activities used in the course can be found at CS-POGIL [31].

3.3.4 Collaboration in control group. For active learning lectures, the instructor encouraged students to work with others at their table to solve the problems after trying them independently. For lab assignments, students optionally collaborated with peers and were required to individually share hypotheses and definitions on the class online forum [19]. For guided inquiry, students formed groups at the beginning of each activity. Groups submitted a single copy of their answers for a participation grade.

3.4 Intervention group

During Spring 2018, the instructor used the intervention, where class sessions consisted entirely of lab assignments and guided inquiry. Table 2 lists the number of class sessions dedicated to each type of learning activity. The instructor replaced active learning lectures with guided inquiry and lab assignments. Students worked in teams (positive interdependence). In addition, for every class period, there was an exit slip that the Reflector responded to (group processing).

3.4.1 Collaboration in intervention group. Students worked on all in-class activities with their assigned team. To accommodate teams in the lab assignments, students shared predictions and outcomes with their team instead of on the class discussion board.

3.4.2 Team organization. Teams were together for four weeks and consisted of 3-4 students. In each week, the instructor rotated the student roles, which gives each student a chance to participate in different ways [7][18]. To form new teams at the end of a 4-week period, the instructor balanced the criteria of mixing prior pairings with student preference. The students picked teams for weeks 13-15 because they doubled as final project teams.

3.4.3 Exit slips. An exit slip consisted of process questions and content questions. The process questions promote group processing. They prompt the student to reflect on how their team is working together and what improvements they might make. The content questions assess the students on key learning objectives and prompt students to formulate questions so that the instructor may give tailored instruction next session. The instructor encouraged the Reflectors to ask teammates for input. Teams received grades for the guided inquiry activities by submitting their exit slips.

3.5 Measures of achievement

3.5.1 Final exams. The final exams had three identical multi-part questions and two different ones. There were two different questions because the control semester covered more content,
specifically pipelining hazards and caches. We only used the three identical questions in our analysis. These questions covered: (1) concepts in architecture and digital logic, (2) design and analysis of microarchitecture and architecture, and (3) applications in digital logic.

The instructor shared the final exam with two colleagues to assess whether it assessed important learning objectives of computer organization. Both colleagues, one who had previously taught the course and the other who had taught a dependent course, approved of the exam.

3.5.2 Concept inventory. Concept inventories are multiple-choice assessments that measure conceptual understanding. The wrong choices are distractors that represent common misunderstandings that students hold. We used the Digital Logic Concept Inventory [32], which has been validated as a post-test of student understanding of concepts in digital logic [33]. The inventory was proctored during the last week of class, before final exams.

3.6 Questionnaire

The pre-test consisted of 15 items on intrinsic value and task value. The post-test consisted of 30 items on intrinsic value, task value, engagement, and peer academic support.

3.6.1 Engagement. Engagement is important to students' learning and achievement in higher education [34]. This construct is multi-faceted, so we included items that measured engagement in instructional activities and perceptions of social support and authenticity of tasks [35].

3.6.2 Intrinsic value. Intrinsic value refers to meaning given to learning tasks, which reflects whether students think what they learn is useful, important, or interesting [36]. To measure intrinsic value, we used items that were adapted from motivational beliefs scales [37].

3.6.3 Task value. Task value is the degree to which students perceive incentives for engaging in learning activities [38]. Students’ task value has been supported as a predictor of choices related to achievement [39]. We asked how helpful each type of course activity was to their learning. The activities were: (abridged) “labs in class,” “labs out of class,” “readings”, “lectures”, “Peer Instruction”, “guided inquiry”, “homework”, and “projects”. To learn why students rated as they did, we also asked on the post-test two free response questions “Why did you rate [activities] as helpful, unhelpful to your learning?” We used open coding [40] to find themes and categorize responses. Two researchers independently formed codebooks, came to consensus on a codebook, independently coded, then came to consensus on codes. They assigned each response between one and four codes.

3.6.4 Peer academic support. Perceived support from teacher and peers may enhance students’ engagement and motivation [41][42]. To measure peer academic support, we used a subscale of four items (e.g., “The students in this class like to help me learn”) that has been associated with higher levels of cooperation [43].

4 RESULTS

4.1 Achievement

To see the effect of the intervention on achievement, we ran a multiple linear regression analysis that controlled for prior achievement using GPA. The model was Achievement ~ Group + GPA. The intervention group scored lower on the final exam than the control group (b= -1.560, t= -2.708, p=0.009) and this difference was statistically significant. We analyzed the three questions (listed in Section 3.5.1) separately. The results were (1) intervention group lower (b= -0.0125, t= -0.049, p=0.961) but not statistically significant, (2) intervention group lower (b= -2.560, t= -2.154, p=0.035) and statistically significant, and (3) intervention group lower (b= -1.615, t= -4.193, p<1e-4) and statistically significant. For DLCI score, the intervention group scored lower (b= -0.452, t= -1.156, p=0.253) but this difference was not statistically significant.

4.2 Engagement

We ran a multiple linear regression analysis to test the effect of the intervention on students' engagement while controlling for group differences. We specifically controlled for students’ intrinsic value of the course, considering that intrinsic value is related to engagement and achievement [44]. The model was Engagement ~ Group + Pre_Intrinsic_Value. The intervention group had lower engagement than the control group (b= -2.934, t= -2.389, p=0.020) and this difference was statistically significant.

4.3 Peer academic support

We ran a multiple linear regression analysis to see the effect of the intervention on peer academic support. Since the task value of in-class collaborative activities at the beginning of the semester is related to orientation with regards to peers, we included it in the equation to control for group differences. The model was Peer_Support ~ Group + Pre_collab_Task_Value. The intervention group had higher peer support (b= 0.046, t= 0.056, p=0.955) but this difference was not statistically significant.

4.4 Task value

4.4.1 Quantitative. We analyzed students’ change in task value of course activities between the pre-test and post-test. We added the task value of labs-in-class, guided inquiry, and Peer Instruction into one construct called collab3 (range 3 to 15). To test the difference between pre and post collab3 (data not normally distributed) we ran a two-tailed Mann-Whitney (µc2-c1=0.43, µi2-i1= -1.0, p < 0.03). Since some students in the intervention may have pragmatically rated Peer Instruction lower because there was none of it in the intervention group, we excluded Peer Instruction from the construct collab2 (range 2 to 10) and repeated the test (µc2-c1=0.23, µi2-i1= -0.70, p < 0.03). We also tested whether the decrease from pre-test to post-test of collab3 and collab2 in the intervention was significant using a two-tailed Wilcoxon signed-rank test (p < 0.03 and p < 0.04). The increases in the control were not statistically significant.
Table 3. Themes for the coded responses to post-test questions “Please explain why you rated the following activities as helpful/not helpful to your learning.”

<table>
<thead>
<tr>
<th>Theme</th>
<th>Why helpful</th>
<th>Why not helpful</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>“Working on the assignments and projects helped me to see how the concepts worked and how to use them.”</td>
<td>Group process “Attempting to take in information while attempting to focus on group interaction is far too distracting to perform either.”</td>
</tr>
<tr>
<td>Learning with peers</td>
<td>“Working on actual code helps cement concepts, and explaining code helps to make you understand it better.”</td>
<td>Unprepared for the work “I would rather prefer lecture type model where the professor discusses all the concepts in class.”</td>
</tr>
<tr>
<td>Feedback</td>
<td>“I usually find that an explanation of why I did something wrong makes it stick the most.”</td>
<td>Help access “…(labs outside of class) lacks the benefit of receiving help and feedback right away.”</td>
</tr>
<tr>
<td>Coherence</td>
<td>“Having organized information relevant to the course material that can then be solidified by doing the homework…”</td>
<td>Difficulty “The projects were very difficult, without time for trial and error, and often the bulk of the project was testing…”</td>
</tr>
<tr>
<td>Instructor help</td>
<td>“We had lots of time to work out what was correct and why, as well as having access to classmates and TAs for help…”</td>
<td>Type of activities “Reading about coding doesn’t help me learn very well, I tend to use the text more as a reference than required reading.”</td>
</tr>
</tbody>
</table>

4.4.2 Qualitative. From responses to why activities were helpful or unhelpful, we identified 17 preliminary codes for helpful and 17 codes for unhelpful, and we grouped these codes into 5 categories for helpful and 5 categories for unhelpful (Table 3). Among the responses to the question asking why the activities were helpful, the category Learning with peers included the subcategory Peer explanation that refers to explanation by someone at the same level of understanding; for example, “sometime classmates’ explanation about the knowledge is easier for me to understand than just listening to the professor.” Students appreciated having access to help from peers, the instructor, and TAs (instructor help). Application included responses that expressed the value of “doing,” “hands-on learning,” and “applying concepts.”

Regarding the reasons why activities were not helpful for learning, students mentioned being unprepared for the work in class. Of these, the subcategory need to be taught before doing includes comments such as “working on labs during class is not very helpful for me because most of us barely understand concept. We are not able to finish assigned work by our own without lecture.” Group process included comments about the challenges students encountered from group dysfunction, including preference for working alone, absent or unprepared group members, lack of social skills, and imbalance of work.

5 Threats to Validity

5.1 External validity

External validity is the extent to which the conclusions of a study generalize to other contexts. We completed this study in a specific context: a specific computer organization course, set in an active learning classroom without a separate lab section, using guided inquiry, with one instructor, and students at one institution. While these control variables assisted the internal validity of the study, they limit generalizability. To support replications or similar studies, we have provided a replication package [45] containing the in-class activities and measurement instruments [46].

Our major curriculum presently has few prior in-class team experiences, so results may vary in departments that introduce cooperative learning experiences earlier [47].

The intervention changed multiple aspects of the course, making it difficult to tease apart the contributions of the independent variables (ratio of activities, presence of various cooperative learning structures). However, themes from our qualitative analysis helped us draw interesting conclusions.

The guided inquiry instruction may not have been representative of POGIL since the instructor had limited experience and had not yet certified the activities with expert POGIL reviewers. We partially mitigated this problem with the Summer 2017 pilot semester.

5.2 Internal validity

Internal validity is the extent to which the conclusions follow from the study. First, the fact that the instructor was on the research team may have biased the experiment. As discussed in 3.2, we took steps to reduce this bias.

Second, instructor graded the exams for each sample at different times, so they may have been graded differently. We mitigated accidental bias by keeping a rubric, notes for interpreting it, and referring to exemplars. Given those safeguards and the lack of an effect in the hypothesized direction, we decided not to regrade the exams as a mixed collection.

Third, the results could be explained by the instructor having more experience with Peer Instruction than guided inquiry (3 prior semesters versus 1). On the other hand, the instructor improved the quality of the inquiry activities between the control and intervention semesters.

Finally, the quasi-experiment was subject to selection bias and response bias due to our drawing of samples from different semesters and students opting into the study. In fact, Fall had a greater proportion of fourth years than Spring (Table 1). Other
work evaluating POGIL in computer architecture courses found that the level of a student affected their response to the pedagogy [48]. We mitigated these biases in our analysis of achievement, engagement, and peer support by controlling for associated factors (Sections 4.1-4.3); however, we did not directly control for class year, so confounding factors may remain.

5.3 Construct validity
Construct validity is the extent to which a study measures what it claims to. Although we used a validated instrument for measuring peer academic support, we unintentionally used a four-point “Often” scale instead of its prescribed five-point “Agree” scale. Fortunately, Cronbach’s alpha was > 0.8, which is in the acceptable range.

Our final exam may have not been a valid measure of achievement in computer organization. While we had other instructors review the exam (Section 3.5), it was not a formal process. We mitigated this weakness by having multiple measures of learning: the final exam and the externally validated DLCI.

The DLCI may not have been a reliable measure of student learning in our course. The reliability was <0.6, indicating students may have made errors due to reasons other than misunderstanding concepts, such as misreading or low incentive to try on an ungraded assessment.

6 Discussion
Regarding RQ1, after controlling for prior achievement the intervention group had lower achievement on two of three final exam questions and no significant difference on the other final exam question and the concept inventory. We attribute the null and negative results partly to the decrease in engagement and task value of collaboration, the subject of RQ2.

Regarding RQ2, after controlling for preliminary intrinsic value, the intervention group had lower engagement. Also, students in the control increased their task value of collaborative activities, while students in the intervention decreased it. Although this result was initially discouraging, we think that there are interesting reasons for it.

First, while the intervention addressed positive interdependence and group processing in ways that the control did not, it may have insufficiently addressed factors of cooperative learning to support the increased amount of guided inquiry. Students in the intervention, which had much more guided inquiry, uniquely said that they needed to be taught before doing. This does not mean students did not like doing; the most common reason for helpfulness was application. Therefore, the instructor could better emphasize the value of the learning cycle of guided inquiry to see gains in task value.

Second, the increased amount of cooperation in the intervention may have lacked critical supports of cooperative learning. Students in the control mentioned being unprepared for in-class activities and students in both conditions mentioned various types of group dysfunction. Dysfunction may also have had a greater effect in the intervention when students had less ability to pick group members. Rotating roles once a week without revisiting them within the same team may have limited the effect of group processing. Finally, lack of individual accountability may be the most likely source of frustration for would-be engaged students. The group process subcategories on attendance, imbalance of work, and preparation in reasons for unhelpfulness suggest that we did not sufficiently address individual accountability. Some ways to increase individual accountability include using peer evaluations [49] and giving individual quizzes in class [11].

Preference for working alone came up seven times, which indicates that student opposition to cooperation, which some researchers feel is left out of cooperative learning models [50], needs to be considered. More favourably, learning with peers was mentioned in 18 why-helpful responses. These included subcategories that have been shown to promote learning: peer explanation [51], help from peers [52][53], and exchange of ideas [54]. This result indicates that plenty of students recognize the benefits of cooperative learning; therefore, we cautiously assert that engagement can be improved if cooperation is comprehensively supported.

7 Conclusion
In a quasi-experiment comparing two offerings of a computer organization course, we found that adding more guided inquiry supported by teams (positive interdependence) and teamwork skills reflections (group processing) lowered some measures of achievement and lowered engagement and task value of collaborative activities. To explain the results, we gathered themes from student explanations for why activities were helpful or unhelpful. The analysis showed that although students commonly valued hands-on learning, some in the intervention also felt they needed to be taught before doing. It also showed that although students commonly valued peer learning, some in both conditions also identified lack of preparation and group dysfunction as unhelpful. These results point to insufficient individual accountability to support the cooperative learning. Our study supports that Computer Organization students recognize some of the benefits of guided inquiry and cooperative learning but that the instructor needs to comprehensively support these strategies for them to succeed over other types of collaborative learning.

Acknowledgments
An ACM SIGCSE Special Projects Award supported the creation of the POGIL activities used in the course. We also thank our anonymous reviewers for thoughtful feedback on improvements.

References