A Multi-institution Exploration of Peer Instruction in Practice

Cynthia Taylor
Oberlin College
Oberlin, OH, USA
cynthia.taylor@oberlin.edu

Jaime Spacco, David P. Bunde
Knox College
Galesburg, IL, USA
{jspacco,dbunde}@knox.edu

Andrew Petersen
University of Toronto Mississauga
Mississauga, ON, Canada
andrew.petersen@utoronto.ca

Soohyun Nam Liao, Leo Porter
University of California at San Diego
La Jolla, CA, USA
{leporter,snam}@eng.ucsd.edu

ABSTRACT

Peer Instruction (PI) is an active learning pedagogy that has been shown to improve student outcomes in computing, including lower failure rates, higher exam scores, and better retention in the CS major. PI's key classroom mechanism is the PI question: a formative multiple choice question on which students vote, then discuss, then vote again. While research indicates that PI questions lead to learning gains for students, relatively little is known about the questions themselves and how faculty employ them. Additionally, much of the work has examined PI data collected by researchers operating in a quasi-experimental setting. We examine data collected incidentally by multiple instructors using PI as a pedagogical technique in their classroom. We look at how many questions instructors use in their courses, the difficulty level of the questions, and normalized gain, a metric that looks at increases in student correctness between individual and group votes. We find normalized gain levels similar to those in existing literature, indicating that students are learning, and that most questions, even those developed by instructors new to PI, fall within recommended difficulty levels, indicating instructors can create good PI questions with little training. We also find that instructors add PI questions over the first several iterations of a new PI course, showing that they find PI questions valuable and suggesting that full development of PI materials for a course may take multiple semesters.

CCS CONCEPTS

• **Social and professional topics** → *Computer science education*;

KEYWORDS

Peer Instruction, Clicker Questions, CS Education, Active Learning

ACM Reference Format:
Cynthia Taylor, Jaime Spacco, David P. Bunde, Andrew Petersen, and Soohyun Nam Liao, Leo Porter. 2018. A Multi-institution Exploration of Peer Instruction in Practice. In *Proceedings of 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE'18).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3197091.3197144

1 INTRODUCTION

Peer Instruction (PI) is an active learning technique in which students respond to multiple choice questions throughout lecture. It was first shown to be highly effective in promoting student learning in the physics community [2, 7]. Within computer science, it has been shown to improve student learning [16, 21, 27, 28], to lower D/F/Withdraw rates and improve major retention [12, 17], and to be effective for both large and small classrooms [8, 15].

While much work within CS has examined PI's effectiveness for promoting student outcomes, little has been done to explore how instructors use it. In this work, we look at a collection of over 4,100 posed clicker questions from 26 different offerings of 10 different courses taught by 7 different instructors at 4 institutions, and focus on 3915 questions by 5 instructors. This data was collected in a natural setting, with instructors using PI as a pedagogical technique with little or no research agenda. As such, the set is a foil for PI data collected in quasi-experimental settings.

Contributions of this work include the collection of a large dataset of clicker questions and student response data, the development of a labeling tool for these questions, and a comparison of this data to previously published results. We begin by examining how much time instructors allow for PI questions and how difficult these questions are. While there are differences in how instructors use PI in their classrooms, we find that the majority of our collected questions are between 35% and 70% correct on the initial vote, within the "good question" difficulty guidelines recommended by Crouch et al [2, 3].

Then, we look at normalized learning gain between solo and group questions. Normalized learning gain has been used to measure student learning from PI questions in prior work [15, 20, 22, 24, 27]. We find normalized gain similar to the results published in these works, indicating that students are learning from PI questions [16, 27].

Lastly, we investigate the questions used in the same course over multiple semesters. We find that instructors increase the number of questions used over time when they are either new to PI or developing a new course. We also see that average learning gains are fairly stable for experienced instructors teaching established courses.

2 BACKGROUND AND RELATED WORK

PI was first developed in physics [2]. Instructors using PI present a mini-lecture introducing a topic, and then ask students a multiple

ITiCSE'18, July 2–4, 2018, Larnaca, Cyprus
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5707-4/18/07…$15.00
https://doi.org/10.1145/3197091.3197144

choice question on the covered concepts. Students first vote individually on the question (the *solo vote*), frequently by using a "clicker", a small electronic device which sends the instructor an aggregated view of the class votes. The students then discuss the question in small groups. Each group comes to consensus on the answer, and then votes again (the *group vote*). Next, instructors facilitate a class-wide discussion on the topic and question, usually showing the class the results of their voting, going over the correct and distractor answers, and answering any student questions. This process is repeated for each topic. Notably, the purpose of the PI questions is student learning rather than assessment; they are either ungraded or graded for participation rather than correctness.

While PI can be implemented using low tech devices such as flash cards rather than clickers [6], the use of clickers allows instructors to collect data on student understanding anonymously and with greater accuracy. If most of the class answers the question correctly in the individual vote, the instructor may skip the group discussion and vote. If most students still answer incorrectly after the group discussion, the instructor may give a broader explanation of the topic. Displaying the histogram of answers to students after the group vote lets students know they are not alone in their misconceptions [5].

PI relies on students preparing for the topic before class. This preparation has traditionally taken the form of students watching a video or reading a book chapter, followed by a quiz on the topic [2]. These quizzes may be administered at the beginning of class using clickers or before class using a web interface [11]. Reading quizzes have been found to be helpful to students in CS classes using PI [11, 26]. However, some CS classes do not use reading quizzes at all [19], or may assign exploratory homeworks instead [4].

PI has been studied in computer science education since 2010 [20, 24]. It has been shown to be well-liked by both students and instructors [13–15, 20]. Using isomorphic questions, several studies [16, 27] have shown that students learn from the discussion phase. PI has been shown to improve student efficacy [25], to improve student retention [12], and to lower fail rates [17]. It has also been used to identify struggling students early in a course [9, 18].

3 METHODOLOGY

In this section we describe how our data was collected and labeled, as well as establishing our research questions.

3.1 Setting

Our study uses over 4,100 question responses from 7 instructors teaching 26 courses at 4 North American institutions over the past 3 academic years, shown in Table 1. Institutions are described in Table 2. All instructors were using iclicker software, which automatically collected student responses and images of questions as a side effect of instructors using PI in their classes.

3.2 Labeling

Each question was labeled using a tool built by one of the authors. This allowed us to: (1) mark solutions, (2) link solo and group votes, and (3) classify the questions into categories.

Here are the labels used to categorize questions:

Paired These are questions asked twice, with a solo vote followed by discussion and group vote. We matched each set of paired votes together, allowing us to compare solo and group votes. In the rest of this work, we count a set of votes as a single PI question.

Table 1: Instructors, their institutions, years of experience using PI, and courses used in this work.

Instructor	Institution	Years of PI	Courses
A	Inst 1	3	CS2
B	Inst 3	5	Computer Systems, Networks
C	Inst 4	6	Computer Architecture
D	Inst 3	5	CS1, Digital Logic, Machine Organization, Computer Systems, Operating Systems
E	Inst 2	2	CS1
F	Inst 2	1	CS1
G	Inst 2	1	CS1

Table 2: Institution type (LAC = Liberal Arts College, RIU = Research-Intensive Institution), class size, and term length.

Institution	Institution Type	Class Size	Term Length
Inst 1	LAC	10-30	9.5 weeks
Inst 2	RIU	100-200	12 weeks
Inst 3	RIU	100-200	15 weeks
Inst 4	RIU	100-400	10 weeks

Single These votes are taken only once, likely because the results indicate that the class already understands the concept or because the instructor failed to close voting between rounds.

Quiz These are votes taken as a reading quiz at the beginning of the period. Unlike the other votes, they are graded for correctness rather than participation.

non-MCQ These non-multiple choice questions represent deliberate use of the clicker for purposes other than standard questions, such as using the timer for in-class discussions or activities, attendance taking with zero-content questions ("Are you here?"), and administrative polls ("When would you prefer the review session?"). We also counted questions as non-MCQ if they relied on information outside of the slide, such as a handout not captured by the software.

From these categories, we count the paired and single votes as *PI questions*. Although single votes do not follow the entire PI process (solo vote, discussion, group vote), some instructors skip the last two parts when the class does very well in the solo vote, indicating the question is too easy to lead to useful peer discussion. Instead, the instructor leads a classroom-wide discussion immediately after the solo vote.

We also marked votes that were not part of a course, such as those triggered by accident, class preparation, or "ghost votes" to help students identify their clickers' unique ID. These were removed from our data and do not appear in our results.

This tool is available for other researchers at:
https://github.com/jspacco/iclickerviewer

3.3 Threats to Validity

This data was collected in a natural setting, and as such suffered from both human and computer error. We have done our best to label clear mistakes as well as other non-PI questions, exclude them from

Table 3: Question type by instructor. * indicates that the instructor reported keeping the poll open across the individual and group votes.

Instructor	Paired	Single	Quiz	Non.MCQ
A	75.7%	14.2%	0.0%	10.2%
B	37.9%	3.2%	45.9%	13.0%
C	83.4%	12.2%	0.0%	4.4%
D	70.5%	1.1%	26.7%	1.7%
E	94.1%	5.9%	0.0%	0.0%
F	22.2%	77.8%*	0.0%	0.0%
G	15.7%	80.6%*	0.0%	3.7%

the resulting analysis, and to include clear explanations in the text when this was not possible. Occasionally the clicker software did not save data correctly: these questions were also labeled as errors. As with any human labeled dataset, it is possible there are flaws in labeling. Finally, data collection was voluntary and instructors who contribute their data may represent "better" instructors (or ones more comfortable with PI).

3.4 Research Questions

This work looks at the following research questions:

(1) How are instructors actually using clickers? What percentage of questions use the entire process (solo vote, followed by group discussion and group vote)?
(2) How many PI questions are used?
(3) How much time do instructors allow for student discussion?
(4) How difficult are the questions that instructors design?
(5) How much do students learn from PI questions, reflected by gain in correct answers between the solo and group vote?
(6) How does instructor use of PI change over time within a course?

These questions were designed to compare our real-world dataset to previous work and advice about PI.

4 RESULTS

In this section we look at how our research questions are answered by this dataset, and compare our results to that of existing work.

4.1 Quizzes, Surveys and Discussions: How Instructors Are Using Clickers

To answer our first research question, we labeled clicker questions using the labels described in Section 3.2. As shown in Table 3, all instructors used clickers for more than just paired questions. In the non-MCQ category, we saw clickers frequently used for course surveys and as a timer for non-multiple choice exercises. Clickers are used for administering reading quizzes by two instructors, both at the same institution, and in one course, make up nearly 50% of the questions. We should note that in all classes, the majority of questions were ungraded, discussion-eligible, questions: this is important as grading clicker questions for correctness appears correlated with lower levels of student satisfaction with PI [14].

We also see a wide variance in the percentage of paired versus single peer instruction questions. The two outliers, instructors F and G report using clicker software in unintended ways due to unfamiliarity with the tool. Instructor G describes his usage thusly: "While I probably didn't push for group discussion enough, I did use it regularly. I tended to push for individual votes quickly, identified an issue to discuss, and then moved to group discussion ... without closing the poll." It is worth noting that while their clicker data over-reports single questions, the students still regularly engaged in the group discussion required for PI. We exclude data from these instructors from further metrics as this makes their data uninterpretable, leaving 3915 clicker questions from 5 instructors.

Table 4: Question statistics for each instructor. We include paired, single, and non-MCQ questions. For paired questions, each pair of questions counts as a single question.

Instructor	# courses	Avg/Course	Avg/Class
A	6	75.3	3.2
B	3	108.3	3.4
C	6	86.3	5.6
D	8	208.9	6.4
E	1	119.0	3.6

Table 5: Number of seconds instructors wait for students to answer PI questions.

Instructor	Avg Solo	Avg Group	Avg Single
A	67.0	110.4	61.1
B	34.4	79.0	38.7
C	65.4	126.2	133.8
D	52.8	65.7	54.8
E	64.6	72.9	51.0

The range of usage for clickers, including quiz questions, single questions, and survey type questions, indicates that researchers may want to dig deeper into self-reported PI classes, since fewer discussion questions may result in less learning gains for students[16, 27].

4.2 Timing and Question Quantity

For our second and third research questions, we looked at the amount of time the clicker software recorded the question as being open for voting, as well as the number of questions recorded in each session. We compare this to the existing research and advice on timing in PI.

Beatty et al. [1] argue for "Question-driven Instruction", in which topics are introduced by questions discussed "a few minutes" before a vote followed by a class-wide discussion, and then optionally a mini-lecture on the topic. They recommend three or four discussion questions per 50 minute lecture. As shown in Table 4, we see three of our instructors use a bit over 3 questions per class and two of them use around 6. This is in the same range as Porter et al. [14], which looked at 7 courses and found 4–7 clicker questions per class.

Table 5 shows the average time each instructor gave for PI questions with a single correct answer. None of the instructors in our data set allow the "few minutes" for discussion recommended by Beatty et al. [1]. Instructors tend to give 30–60 seconds for the solo vote and 1–2 minutes for the group discussion.

In most cases, instructors gave the same or less time to single votes than to solo votes followed by group discussion. This may reflect single votes being easier questions, as is shown in Table 7, where the average percent correct for single votes is higher than the percent correct for a solo vote in every case. In the case of C, where the solo vote is given longer, this reflects a number of questions that were

Table 6: Fraction of easy, medium, and hard questions by instructor, using difficulty levels from Smith et al. [23]. Includes both solo votes and single votes.

Instructor	Easy >70%	Medium 35-70%	Hard <35%
A	15%	49%	36%
B	21.2%	51.9%	26.8%
C	10.1%	51.2%	38.7%
D	21.4%	59.3%	19.3%
E	45.4%	35.3%	19.3%

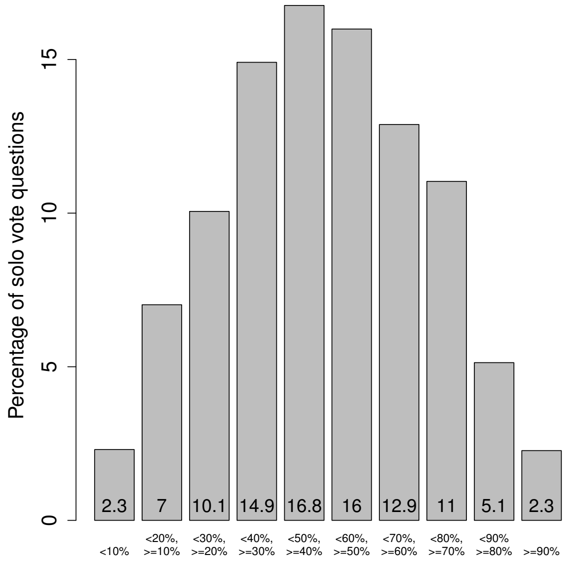

Figure 1: Percent correct on the solo vote of paired questions and single questions, binned by deciles.

lengthy to work out, but had one clear correct answer (e.g. tracing data references in a cache), which were given as single questions.

4.3 Question Difficulty

To answer our fourth research question on how difficult instructor-developed PI questions are, we look at the percentage of students who correctly answer the first vote, a difficulty metric which is often used within existing PI literature. There is no hard and fast rule on what percentage is best, but Crouch et al. [2, 3] recommend between 35% and 70% correct on the initial vote, and Smith et al. [22] recommend below 80%. Zingaro and Porter [27] divide questions into "easy" and "difficult", where hard is below 50% correct on the initial vote, and easy is above 50%. They argue for more difficult questions, pointing out that in their study, difficult questions lead to a 25% student improvement on isomorphic questions after group discussion, and 42% after instructor lead discussion, suggesting that more difficult questions provide the greatest student learning gains.

Smith et al. [23] establishes cutoffs for easy, medium, and hard questions, where easy is more than 70% correct on the first vote, medium is 35–70% correct (roughly aligning with Mazur's guideline), and hard is less than 35%. These cut offs are also used by Porter et al. [15, 16] and Simon et al. [20]. Table 6 shows that for most instructors, above 50% of the questions fall in the medium, good question range, indicating that instructors, even when beginning PI, develop clicker questions with the suggested difficulty level.

Table 7: Percentages of correct answers for each vote of a paired question, and for single questions.

Instructor	Paired, Solo Vote	Paired, Group Vote	Single Vote
A	41.3%	55.7%	70.8%
B	49.6%	61.8%	75.1%
C	40.5%	53.5%	63.6%
D	53.3%	62.9%	61.9%
E	60.5%	70.8%	72.1%

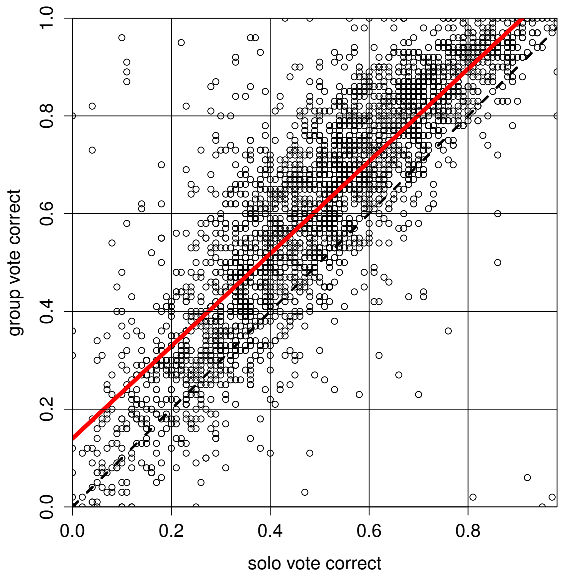

Figure 2: Paired questions, with solo vote correctness on the x-axis, and group vote correctness on the y-axis. The dashed line is the xy-axis (points above this line show gains from solo to group vote), and the solid line is the regression trend line.

Figure 1 shows percent correct for the initial, solo vote of paired questions, binned by decile across all instructors. We see a clear normal distribution centered around the 50% percent correct range. The vast majority of the questions fall in 35-75% recommended range.

Table 7 shows question correctness broken down by solo, group and single vote. We see that single votes have the highest percent correct, which follows the intuition that instructors will omit group discussion on questions that a large majority of their students already get correct. We also see that the group vote is higher than the solo vote, indicating that students are learning from group discussion.

4.4 Normalized Learning Gains

Our fifth research question asks how student correctness changes from the solo vote to the group vote. Figure 2 provides each paired question plotted with the percent correct in the solo vote as the x-axis, and the percent correct on the group vote as the y-axis. First, as one might expect, we see that there is a strong correlation between

Table 8: Normalized Gain (NG) for easy, medium, and hard questions, broken down by instructor.

Instructor	Easy Solo	Easy Group	Easy NG	Med Solo	Med Group	Med NG	Hard Solo	Hard Group	Hard NG	Overall NG
A	79%	73%	0.15	51%	68%	0.35	21%	36%	0.13	0.26
B	79%	91%	0.65	53%	68%	0.32	23%	32%	0.05	0.31
C	76%	85%	0.41	50%	63%	0.28	23%	36%	0.15	0.24
D	79%	87%	0.42	52%	63%	0.24	24%	31%	0.03	0.25
E	82%	89%	0.37	51%	63%	0.26	27%	42%	0.21	0.30

the two, with most points clustering around a line slightly above center. Second, we see that most questions have students perform better on group vote than solo (as we might hope).

To aggregate the results of multiple questions, we use two measures of this improvement. *Raw learning gain* (RG) for a single question is the difference between the percents correct on the solo and group votes. Since the RG achievable on a question depends on the correctness of the solo vote, *normalized learning gain* (NG) is commonly used instead [15, 20, 24]. We use the version of the NG metric defined in Marx and Cummings [10], shown below:

$$NG = \begin{cases} 100 \times \frac{group - solo}{100 - solo} \text{ if group > solo} \\ 100 \times \frac{group - solo}{solo} \text{ if group} \leq \text{solo} \end{cases}$$

When averaging the gain over multiple questions, we compute NG per question and then average the results. Normalized gain scales the difference between group and solo vote by the percentage of students who answered the initial vote incorrectly, allowing for comparisons of improvement between both hard and easy questions. However, it may penalize hard questions, as hard questions need more improvement to achieve the same normalized gain as easy questions. To avoid this, we compare the normalized gain of easy, hard and medium questions separately, as well as providing a single normalized gain metric for comparison with other work.

Table 8 shows normalized gain broken down by the instructor. We see that in general, hard questions show a lower normalized gain, as expected for the metric. In the CS PI literature, Porter et al. [15] report a NG of 0.65 for easy, 0.55 for medium and 0.31 for hard. Zingaro [24] report a 0.29 NG for a remedial CS course, Simon et al. [20] report a 0.41 NG for CS1 and 0.35 NG for CS 1.5 [20], and Zingaro and Porter [27] report a 0.44 NG for CS1 (0.34 for hard questions, and 0.49 for easy). While our normalized gain is slightly lower, it is in the range of these reports. Previous work has used isomorphic questions to show that normalized gain reflects student learning in the discussion section [16, 27]. Normalized gain in the same range as previously reported work is a promising sign that students are learning from peer instruction "in the wild", not just in the context of courses being taught by researchers studying peer instruction.

4.5 Instructor Use of Peer Instruction Over Time

For our sixth research question, we consider how many PI questions instructors ask in classes that they have taught multiple times. Tables 9, 10, and 11 give question statistics for multiple offerings in three different situations.

Table 9: Question statistics for six consecutive offerings of CS 2 at an American liberal arts college by a new PI instructor.

CS2	# CQs	Avg/Class	Avg NG	Avg RG
Winter 15	60	3.2	0.14	0.08
Spring 15	72	3.3	0.28	0.15
Winter 16	70	2.9	0.27	0.15
Spring 16	77	2.7	0.27	0.16
Winter 17	87	3.3	0.26	0.13
Spring 17	86	3.7	0.31	0.19

Table 10: Question statistics for four consecutive offerings of Computer Systems at a large American public university. An instructor experienced with PI but new to this course designed the slides. They taught all offerings except Spring 16, when a different instructor used the slides. Fall 16 is the only version this course offered MWF rather than TuThr.

Computer Systems	# CQs	Avg/Class	Avg NG	Avg RG
Fall 15	194	4.8	0.22	0.09
Spring 16	143	5.5	0.30	0.11
Fall 16	236	5.6	0.19	0.08
Spring 17	211	8.1	0.17	0.08

Table 11: Question statistics for six consecutive offerings of Computer Architecture at a large American public university by an experienced PI instructor. They taught fewer meetings in Fall 14.

Computer Architecture	# CQs	Avg/Class	Avg NG	Avg RG
Fall 14, Sec. 1	74	5.3	0.21	0.11
Fall 14, Sec. 2	76	5.4	0.23	0.13
Fall 15	93	5.8	0.20	0.11
Fall 16	96	5.6	0.28	0.15
Spring 17, Sec. 1	94	5.5	0.22	0.13
Spring 17, Sec. 2	85	5.7	0.29	0.15

The data in Table 9 comes from an instructor newly adopting PI; the first offering is their second term using PI. The number of questions increases fairly steadily as the course is converted more completely to the new pedagogy. The data in Table 10 comes from an instructor experienced with PI but for whom the course is a new prep. Again, the number of questions increases; the decline in Spring 16 was when another instructor taught the course. Finally, the data in Table 11 come from an instructor experienced both with PI and this course; Fall 14 was their 6th offering. The number of questions varies here too, but this is largely explained by the instructor teaching fewer classes in Fall 14 due to significant travel that term. Instead, we look at the number of questions per class, which is quite steady, suggesting that the instructor has reached their desired level of questions while the other instructors are still converting the class into the PI format.

Although based on only a few instructors, this suggests that instructors new to PI convert their courses to the new format over a period of multiple terms, (note that Avg/Class is still significantly lower for the first instructor than the other two) and that even instructors experienced with PI may spend several terms converting a new course.

While that message may sound discouraging, Table 9 tells a very positive story about adopting PI. In Winter 15, the second term in which this instructor uses PI, they have an average NG of 0.14. However, the following term it doubles to 0.28, and then stays consistently between 0.28 and 0.31. We see this as an instructor who becomes proficient with PI and then consistently uses it well within only two terms. Thus, proficiency with the technique comes relatively quickly even though the course continues to evolve. Also, once proficiency is achieved, all the instructors see a fairly steady average NG.

Overall, Tables 9–11 seem to show that course-level average gain (RG and NG) is fairly steady for repeated offerings of a course once the instructor has become familiar with PI, but it would be interesting to see how stable the gain is at lower granularity (e.g., at the lecture or the question level). Also, different instructors stabilized at different gain levels, which begs the question of whether the differences are caused by how instructors used PI, the course material, the level of the course in the curriculum, or aspects of the student population.

5 CONCLUSION AND FUTURE WORK

In this work, we look at a labeled collection of over 4,000 posed PI questions collected by 7 different instructors over 3 years as a side effect of using PI in their teaching. We use this dataset to examine how instructors use PI in their classes, and how this usage compares to existing literature and best practices. We find that the majority of instructor questions fall within recommended difficulty levels, and that student learning gains between the solo and group vote are similar to those previously reported. We see evidence that new adopters can deploy PI effectively within two terms. Lastly, we find that in repeated courses, instructors include more PI questions over time. Overall, we find that PI "in the wild" behaves similarly as in the quasi-experimental setting where it was first examined by researchers.

This paper offers a first look at this dataset. We plan to investigate this data further, including research into how instructor use of PI questions changes over repeated course offerings, common PI question themes, and characteristics of successful PI questions.

ACKNOWLEDGEMENTS

Thank you to everyone who contributed data. Researchers who are interested in contributing clicker data for future work, or using our dataset in their projects are encouraged to contact us.

REFERENCES

[1] Ian D. Beatty, William J. Gerace, William J. Leonard, and Robert J. Dufresne. 2006. Designing effective questions for classroom response system teaching. *American Journal of Physics* 74, 1 (2006), 31. https://doi.org/10.1119/1.2121753 arXiv:physics/0508114

[2] Catherine H Crouch and Eric Mazur. 2001. Peer Instruction: Ten years of experience and results. *American Journal of Physics AIP Conference Proceedings* 69, 9 (2001), 970–977. https://doi.org/10.1119/1.1374249ÍŤ

[3] Catherine H Crouch, Jessica Watkins, Adam P Fagen, and Eric Mazur. 2007. Peer instruction: Engaging students one-on-one, all at once. *Research-Based Reform of University Physics* 1, 1 (2007), 40–95.

[4] Sarah Esper, Beth Simon, and Quintin Cutts. 2012. Exploratory homeworks: an active learning tool for textbook reading. In *Proceedings of the Ninth Annual International Conference on International Computing Education Research.* 105–110.

[5] Jennifer K Knight and William B Wood. 2005. Teaching more by lecturing less. *Cell Biology Education* 4, 4 (2005), 298–310.

[6] Nathaniel Lasry. 2008. Clickers or flashcards: Is there really a difference? *The Physics Teacher* 46, 4 (2008), 242–244.

[7] Nathaniel Lasry, Eric Mazur, and Jessica Watkins. 2008. Peer instruction: From Harvard to the two-year college. *American Journal of Physics* 76, 11 (2008), 1066–1069.

[8] Soohyun Nam Liao, William G. Griswold, and Leo Porter. 2017. Impact of Class Size on Student Evaluations for Traditional and Peer Instruction Classrooms. In *Proceedings of the ACM SIGCSE Technical Symposium on Computer Science Education.* 375–380.

[9] Soohyun Nam Liao, Daniel Zingaro, Michael A. Laurenzano, William G. Griswold, and Leo Porter. 2016. Lightweight, Early Identification of At-Risk CS1 Students. In *Proceedings of the 2016 ACM Conference on International Computing Education Research.* 123–131.

[10] Jeffrey D Marx and Karen Cummings. 2007. Normalized change. *American Journal of Physics* 75, 1 (2007), 87–91.

[11] Roy P Pargas and Dhaval M Shah. 2006. Things are clicking in computer science courses. In *ACM SIGCSE Bulletin*, Vol. 38. 474–478.

[12] Leo Porter, Cynthia Bailey Lee, and Beth Simon. 2013. Halving Fail Rates Using Peer Instruction: A Study of Four Computer Science Courses. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education.* 177–182.

[13] Leo Porter, Cynthia Bailey Lee, Beth Simon, Quintin Cutts, and Daniel Zingaro. 2011. Experience report: a multi-classroom report on the value of peer instruction. In *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education.* 138–142.

[14] Leo Porter, Dennis Bouvier, Quintin Cutts, Scott Grissom, Cynthia Lee, Robert McCartney, Daniel Zingaro, and Beth Simon. 2016. A Multi-institutional Study of Peer Instruction in Introductory Computing. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education.* 358–363.

[15] Leo Porter, Saturnino Garcia, John Glick, Andrew Matusiewicz, and Cynthia Taylor. 2013. Peer Instruction in Computer Science at Small Liberal Arts Colleges. In *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education.* 129–134.

[16] Leo Porter, Cynthia Bailey Lee, Beth Simon, and Daniel Zingaro. 2011. Peer Instruction: Do Students Really Learn from Peer Discussion in Computing? *Proceedings of the Seventh International Workshop on Computing Education Research* (2011), 45–52.

[17] Leo Porter and Beth Simon. 2013. Retaining Nearly One-third More Majors with a Trio of Instructional Best Practices in CS1. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education.* 165–170.

[18] Leo Porter, Daniel Zingaro, and Raymond Lister. 2014. Predicting Student Success Using Fine Grain Clicker Data. In *Proceedings of the Tenth Annual Conference on International Computing Education Research.* 51–58.

[19] Beth Simon and Quintin Cutts. 2012. CS Principles Pilot at University of California, San Diego. *ACM Inroads* 3, 2 (2012), 61–63.

[20] Beth Simon, Michael Kohanfars, Jeff Lee, Karen Tamayo, and Quintin Cutts. 2010. Experience report: peer instruction in introductory computing. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education.* 341–345.

[21] Beth Simon, Julian Parris, and Jaime Spacco. 2013. How we teach impacts student learning: peer instruction vs. lecture in CS0. In *Proceeding of the 44th ACM technical symposium on Computer science education.* ACM, 41–46.

[22] MK Smith, WB Wood, Ken Krauter, and JK Knight. 2011. Combining peer discussion with instructor explanation increases student learning from in-class concept questions. *CBE-Life Sciences Education* 10, 1 (2011), 55–63.

[23] M. K. Smith, W. B. Wood, W. K. Adams, C. Wieman, J. K. Knight, N. Guild, and T. T. Su. 2009. Why peer discussion improves student performance on in-class concept questions. *Science* 323, 5910 (2009), 122–124. arXiv:gr-qc/0208024

[24] Daniel Zingaro. 2010. Experience Report: Peer Instruction in Remedial Computer Science. In *Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications(Ed-Media).* University of Toronto.

[25] Daniel Zingaro. 2014. Peer instruction contributes to self-efficacy in CS1. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education.* 373–378.

[26] Daniel Zingaro, Cynthia Bailey Lee, and Leo Porter. 2013. Peer instruction in computing: the role of reading quizzes. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education.* 47–52.

[27] Daniel Zingaro and Leo Porter. 2014. Peer Instruction in computing: The value of instructor intervention. *Computers & Education* 71 (2014), 87–96.

[28] Daniel Zingaro and Leo Porter. 2015. Tracking Student Learning from Class to Exam Using Isomorphic Questions. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education.* 356–361.