Halving Fail Rates using Peer Instruction: A Study of Four Computer Science Courses

Leo Porter
Skidmore College
Saratoga Springs, NY 12866
Leo.Porter@skidmore.edu

Cynthia Bailey-Lee and Beth Simon
University of California, San Diego
La Jolla, CA 92093
{clbailey, bsimon}@cs.ucsd.edu

ABSTRACT
Peer Instruction (PI) is a teaching method that supports student-centric classrooms, where students construct their own understanding through a structured approach featuring questions with peer discussions. PI has been shown to increase learning in STEM disciplines such as physics and biology. In this report we look at another indicator of student success – the rate at which students pass the course or, conversely, the rate at which they fail. Evaluating 10 years of instruction of 4 different courses spanning 16 PI course instances, we find that adoption of the PI methodology in the classroom reduces fail rates by a per-course average of 61% (20% reduced to 7%) compared to Standard Instruction (SI). Moreover, we also find statistically significant improvements within-instructor. For the same instructor teaching the same course, we find PI decreases the fail rate, on average, by 67% (from 23% to 8%) compared to SI. As an in-situ study, we discuss the various threats to the validity of this work and consider implications of wide-spread adoption of PI in computing programs.

Categories and Subject Descriptors
K.3.2 [**Computer Science Education**]

General Terms
Performance, Human Factors.

Keywords
Peer Instruction, Active Learning, Student-Centric Learning, Assessment.

1. INTRODUCTION
Peer Instruction (PI) is a teaching method – an instructional approach applicable to a range of disciplines and content. PI supports a student-centered learning environment by replacing some of lecture time's traditional "sage on the stage" activity with "guide on the side" student-focused activity. That is, instructor explanation is augmented and/or replaced with carefully crafted questions designed to engage students' learning. The "peer" in PI comes from the fact that core to the PI methodology is having students discuss and analyze questions in small groups. While many of us may ask students to "work in groups" at times, PI generally uses clickers to motivate full participation and to enable quick gathering and review of the entire class' views. The combination of personal responsibility (perhaps by assigning points for clicking), peer discussion, and student feedback (including how their peers are doing) underlie the PI experience. As one of a number of student-centric learning approaches (including collaborative learning, active learning, cooperative learning and guided inquiry learning, problem-based learning [4]), PI has appeal because it can be implemented in "standard" lecture halls and with relatively fewer course modifications than other techniques (e.g. problem-based learning).

There is a significant body of research showing the positive impact of PI on learning gains, most notably in physics (where performance on a standardized concept inventory improved two-fold) [2], but also in chemistry and biology [3,6,7]. Numerous calls have been made in the U.S. encouraging university faculty to adopt such evidence-based practices–including in reports from the National Research Council and the President's Council of Advisors on Science and Technology [12]. Such efforts are considered critical to increasing the populace's preparation in STEM areas.

In support of this call, PI has been adopted at our institution in a variety of computing courses by various instructors over the past four years. Here we perform a *post-hoc, in-situ study of the adoption of PI in four different courses (CS1, CS1.5, Theory of Computation, and Computer Architecture), spanning 16 course instances, taught by seven different instructors.* Due to the lack of standardized assessments and the nature of the in-situ study (instructors retain academic freedom, set their own exams, etc.), it is not possible to report strict learning gain performance. However, in this report we seek to consider the impact of PI from a programmatic point of view. Specifically we seek to answer the question: *Do fewer students in PI classes have to retake the course due to withdrawing, or earning a D or F grade (WDF) than in standard instruction courses?*

Obviously, there are many benefits to more students passing a course – reduced time to degree, smaller class sizes (assuming students re-take the course to pass it), possibly increased retention in the major, and even reduced student and instructor angst.

The results from our study are very encouraging:

- Course fail rates are reduced 25-81% — with an average (by course) reduction of 61% (z-test, $p<0.01$).
- It is not simply that instructors who adopt PI are "better" than others who teach that course. We measured the change in fail rate for four instructors who had taught the course using standard instruction (lecture) before adopting PI. In this within-instructor study, fail rates are reduced by 40-87% with an average (by instructor) reduction of 66%.
- Finally, in a 10-year retrospective of one specific course, CS1, we find a) the *average* fail rate in non-PI offerings

(N=18) is higher than the fail rate in *any* PI offering (N=9) and b) the average fail rate in PI offerings is lower than all but one instance of non-PI offerings.

Throughout this paper, the *fail rate* refers to the number of students earning a W (withdraw), D, or F grade out of the total number of students earning a passing (A,B,C) or failing (W,D,F) grade.

2. BACKGROUND - PI

The PI methodology commonly consists of the following components: [2]:

1. **Before class.** Students are assigned to complete preparatory work (often textbook reading, but this could include watching online lectures). The goal is to have students learn some of the more basic items, concepts, or definitions before class, so that these do not have to be presented in class – creating time for student engagement. To incentivize students to complete this work, a quiz or other assessment of some sort is given before each "lecture" (either online, or perhaps at the beginning of the lecture [2,7,15].
2. **During class.** Students are posed questions designed to help them confront and explore challenging concepts and issues. Often these questions are posed as multiple-choice and students gain credit for answering the questions with a clicker (or other electronic polling device). Specifically, the algorithm of a clicker question should be:
   a. Pose a question, students answer individually (generally, results not displayed for class).
   b. Small group discussion (2-3 students) where students discuss their thinking and share their analyses with each other.
   c. Students all answer a second time, perhaps changing their answer based on group discussion. (The results of student responses can be shown at this point, or after the discussion below.)
   d. Class-wide discussion led by the instructor, but preferably first asking students to share the explanations and discussions they had in their group. The instructor provides clarification or a model how the question can be analyzed. The correct answer is clearly indicated.

Traditional explanatory lecture materials are likely reduced or removed (e.g. basic materials to be learned in pre-class work). However, lecture-style explanation or examples may either precede or follow clicker questions to provide further explanation or to clarify challenging issues or concepts from the pre-class preparation.

Colloquially, instructors implementing the PI methodology are often described as using clickers in their class. This is not always the case. Clickers are a technology that can be used in many ways. In this paper we report specifically on the use of clickers to implement the PI methodology[1].

[1] Although the PI pedagogy may be implemented with lower-tech alternatives (e.g. colored cards), research is mixed on the effectiveness. A large study in biology showed an advantage for clickers over cards [5], though a smaller physics study showed no benefit [7]. In any case, clickers are bought by students like textbooks and are generally re-sellable.

3. RELATED WORK

Peer Instruction is an active learning pedagogy that utilizes the lecture time to engage students in problem solving and group discussion. PI was introduced by physics professor Eric Mazur [2] and has been extensively studied in university physics classrooms. It has been adopted in other natural sciences [3,6,7]. PI in physics courses has been shown to significantly increase student learning, doubling the normalized learning gains [2].

PI use is associated with lower failure rates in introductory biology classes [6]. In physics classes using PI, a factor of two to three reduction in mid-course withdrawals was reported [8].

In recent years, PI has begun to be adopted in computer science [4,9,10,11,15,18]. In computer science, PI has been shown to be valued by students [10, 15], be valued by instructors [10], and result in individual learning [11]. We are not aware of any study of the impact of PI on pass/fail rates in computer science courses.

4. METHODS

This work reports fail rates, that is, incidence of students withdrawing from the class or completing the course with a grade of D or F. The study draws data from the previous 10 years for four courses, encompassing 120 classes, taught by more than 20 different instructors, and a total enrollment of more than 10,000.

4.1 Institution

All results reported are for courses taught at a large, public, research-oriented institution. The academic year at this institution consists of three 10-week quarters. All summer session classes were excluded from this study, because the student and instructor demographics diverge from the regular school year, and because summer sessions are run on an intensive five-week schedule.

At this institution, students may drop classes up to four weeks into the term without any record. After the fourth week and until the end of the ninth week, students may drop classes with a W ("withdraw") grade recorded. All others completing the course either earn a passing grade (A, B, C, P (pass)) or a failing grade (D, F, NP (no pass)).

4.2 Courses and Instructors

We collected all enrollment history and grade data for the past 10 years for four courses that have been taught using PI at this institution. All course instances not using PI are reported as standard instruction (SI) courses. The numbers of SI instances and PI instances of each course occurring in those years are shown in Table 1. The "Total Enrollment" figure in Table 1 is not the number of distinct students, because many students took more than one of the four classes.

Most results are reported in aggregate by course, combining classes taught by different instructors. We have additionally separated out data for courses that have both SI and PI instances for the same instructor. We will identify the four instructors in this category by number (1-4). Of these, one instructor taught Computer Architecture and the other three taught CS1.

Statistical significance was determined using a two-tailed z-test for two independent groups, comparing the population of students impacted by SI against the population impacted by PI. Statistical significance is denoted as a p-value of less than 0.05.

4.3 Instructor Background

The instructors reflect a range of career stages and history with PI. Of the seven who used PI, two are active in research about PI and mentoring others in the practice, and developed their own PI

Table 1. Profile of classes studied. Instructor Totals may not reflect sum as instructors may teach more than one course.

		CS1	CS1.5	Theory	Arch.	Total
Number of Classes	SI	18	23	31	32	104
	PI	9	4	2	1	16
Number of Different Instructors	SI	5	5	9	9	23
	PI	4	1	2	1	7
Total Enrollment	SI	1764	1937	2674	2237	8612
	PI	1296	361	343	68	2068

materials. The five others used materials developed by others. Some had little or no exposure to PI until immediately before the start of the quarter. All of the following career titles and ranks are represented amongst both the SI and PI instructor pools in this study: teaching-track faculty, tenure-track or tenured faculty, research scientist adjunct faculty, temporary adjunct faculty, and graduate students.

4.4 Threats to Validity

This study is a post-hoc, in-situ study of independently run university classes and, by that nature, cannot impose experimental controls. We identify various threats to the validity of this work, and in some cases, efforts made to manage these threats. Readers are encouraged to consider these as they evaluate likeliness of replication of the results.

Variation in Difficulty between Instances of the Same Course. As a post-hoc study, no controls were imposed to ensure that instructors did not make their classes easier (which could impact W rate), or their final exams and grades more lenient (which could impact D/F rates). Instructors retain academic freedom over their classes, and determine assignments, exams, and grades. Even for our Instructors 1-4, for whom we have both SI and PI data, we cannot assume that exam and grading difficulty was constant between classes, as PI might change an instructor's perspective on what is fair. On the other hand, students have continued in our program and there has been no noticeable increase in "drop outs" or poorer performance in follow-on courses, though this has not been formally measured.

High-Quality Instructors Self-Select to Use PI. Perhaps the population of instructors who would adopt a new teaching practice, such as PI, self-selects for instructors who are better overall. For this threat, we present two mitigating factors. First, three of the seven PI instructors (two in CS1 and one in theory) did not adopt PI in a way characteristic of pure self-selection. They were assigned to a class within a week or two of the start of the term and they adopted PI, in part, because PI materials were available for the course and there was no time to prepare anything else. Second, we report intra-instructor results for Instructors 1-4, who taught the same course in both SI and PI modes. These results therefore control for general quality of the person as an instructor (though, as noted earlier, changes in grading difficulty are still possible).

Variation in Other Aspects of Course Design. Due to principles of academic freedom, instances of a given course can vary, sometimes widely, in course design and even in the topics covered. In particular, we note that CS1 was changed in several significant ways at the same time that it was switched to PI– adopting pair programming for assignments and a contextualized media computation approach similar to those reported in [10,14]. These changes apply to all CS1 PI instructors.

Reproducibility of Outcomes Depends on Quality of PI Materials. The materials for two courses were developed by an instructor with a background in computing education research and who had notable experience in best-practices uses in physics. Both of the other courses had materials developed by instructors who TAed for or were otherwise advised by the first instructor.

Students may have Changed over Time. The first PI course at our institution occurred in the Fall of 2008. Hence, when comparing PI vs. SI, many of the PI classes were taught later in our 10 year window than the SI classes. The decrease in fail rate may be partially attributable to a potential improvement in our students. We did analyze fail rates for the past 5 years (rather than 10 years) and found similar results. In Section 5.4, we extend this analysis by evaluating another introductory CS course to determine if a change occurred between the SI and PI periods of study and no statistically significant change is found.

5. RESULTS

We look at the impact of PI adoption in three ways: a longitudinal review of fail rates in PI and SI classes in the Fall 2001-Spring 2012 timeframe, a within-instructor comparison of fail rates for those who taught a course at least once using each method (PI and SI), and a detailed, per term, longitudinal look at fail rates in CS1 (PI course N=9, SI course N=18).

5.1 Longitudinal Effects

In Figure 1, we provide the reduction in fail rate for the course. We define reduction in fail rate as:

$$\boldsymbol{Reduction = (Fail\_Rate_{SI} - Fail\_Rate_{PI}) / Fail\_Rate_{SI}}$$

We can see from Figure 1 that the PI adoption reduces fail rates dramatically (59% or more) in 3 of the 4 courses with statistically significant improvement ($p<0.05$) in all but the CS1.5 course ($p=0.066$). There are a number of issues that might explain why PI had less of an impact in CS1.5 – including that it is a direct follow-on of CS1 – which are further explored in the Discussion.

It is also of value to look at the raw fail rates of students in both SI and PI courses. Figure 2 provides these results. Somewhat surprisingly, Theory edges out CS1 for highest fail rate in the SI setting. PI reduces fail rates in the lower division courses to around 10% and in upper division courses to around 5%. Particularly noteworthy is the reduction in fail rate in the theory course. In general, this course has a reputation of being a challenging and not-well-loved course at our institution – both because of its math-based nature and because students do not generally feel it relevant to their future careers. Exactly what aspects of PI might be contributing to this increased success and why is a subject for future work.

5.2 Within-Instructor Effects

A potential threat to validity is that those who would choose to adopt PI are "excellent" instructors. To partially address this concern, we compared the fail rates of four instructors who had taught the same course using both SI and PI methods.

Figure 3 shows reduction in fail rates and Figure 4 provides absolute fail rates for these four instructors. All instructors experience a statistically significant ($p<0.05$) reduction in fail rate. Averaged by instructor, fail rate is reduced by 15% by switching to PI—a reduction of 67%. The instructor who originally adopted

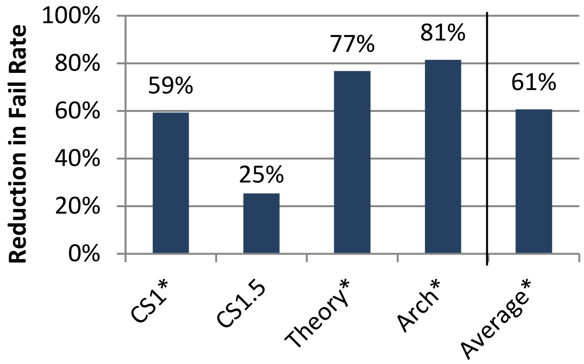

Figure 1. Reduction in Fail rates by course. Changes marked with a * are statistically significant (p<0.05).

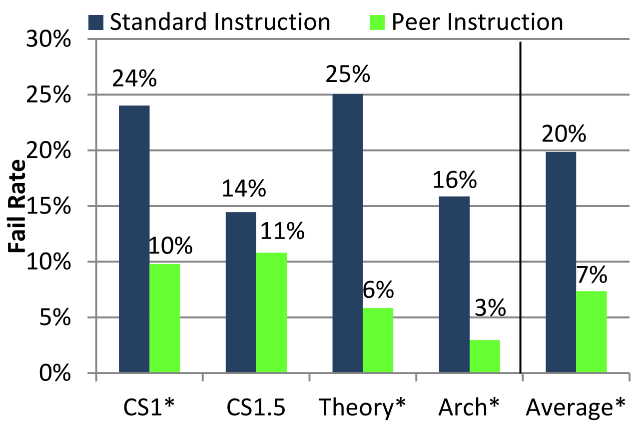

Figure 2. Comparison in Fail Rates in SI and PI course offerings. Changes marked with a * are statistically significant.

PI at our institution and has served as a mentor for many of the other adopters is Instructor 1. Oddly, that instructor experiences one of the smallest benefits, 10%. Instructors 2 and 4 were mentored in the PI adoption process by Instructor 1 with Instructor 4 receiving the most support. The improvement by Instructor 4 bodes well for other faculty looking to adopt PI and may emphasize the value of mentoring. It bears noting that no instructor who taught with PI in a given course has subsequently returned to SI for that course.

5.3 Further Analysis: CS1

CS1 has the largest number of PI adopters (4) and PI sections (9), which makes looking at variations across SI and PI instances more meaningful. Figure 5 provides the fail rates per section of CS1 arranged with SI on the left and PI on the right, and within method, sorted by fail rate. The overall average for SI and PI sections is on the far right. The number of students is provided as well is the term: those without a + were taught in the fall, those with a + were taught in the Winter or Spring. Winter and Spring classes commonly have a different student demographic (fewer majors, etc.) than Fall classes.

Of interest is that although the Winter/Spring sections were among the worst instances with SI, this does not appear to be the case for PI instances. Another point of interest is that class size

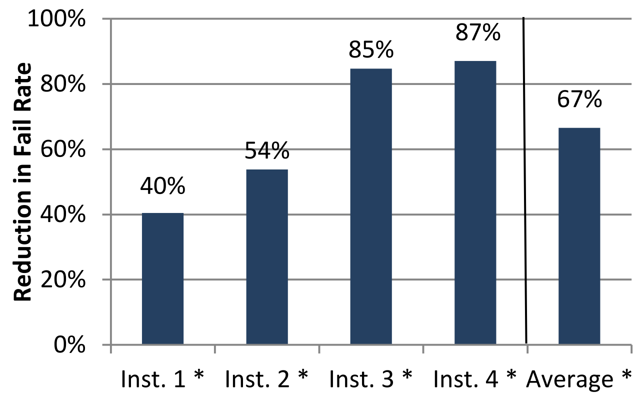

Figure 3. Reduction in Fail rates for instructors who taught both SI and PI instantiations of the same course. Changes marked with a * are statistically significant.

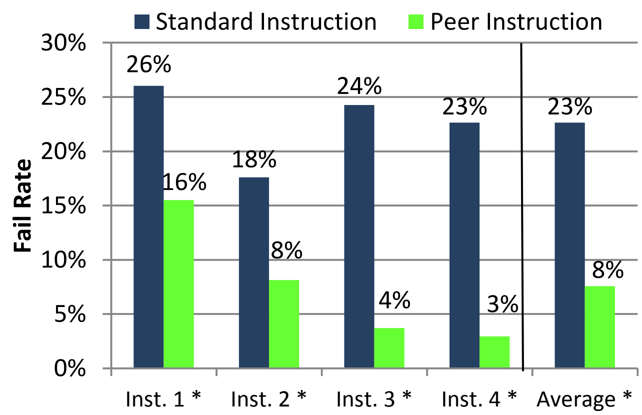

Figure 4. Fail rates for the 4 instructors who taught both SI and PI instantiations of the same course. Changes marked with a * are statistically significant.

does not seem to contribute to fail rate trend – in either instructional mode. PI again stands out as providing a significant reduction in fail rates. In fact, the *worst* PI instance has a lower fail rate than the *average* SI instance.

5.4 Student Improvement

One challenge to the validity of the study might be that students at our institution have been getting stronger (as evidenced by their ability to pass courses) over the past few years. To help evaluate this possibility we evaluated the fail rates for a different advanced-track CS1 course (for those with prior programming experience). The fail rates in the advanced track CS1 for more recent instances (Fall 2008- present) were only slightly lower than older instances (Fall 2001-Spring 2007) (21% vs. 20%) and are not significantly different (p=0.54). Although the course makeup may be different for each track of CS1, this provides some contradictory evidence to the claim of students improving with time. Additional analysis in Theory and Architecture do not evidence notable differences in SI fail rate averages when considered for the last 10 years or just the last 5.

5.5 Statistical Significance

Table 2 provides the statistical significance of the evaluations in fail rates reported above using a two-tailed z-test for two independent groups.

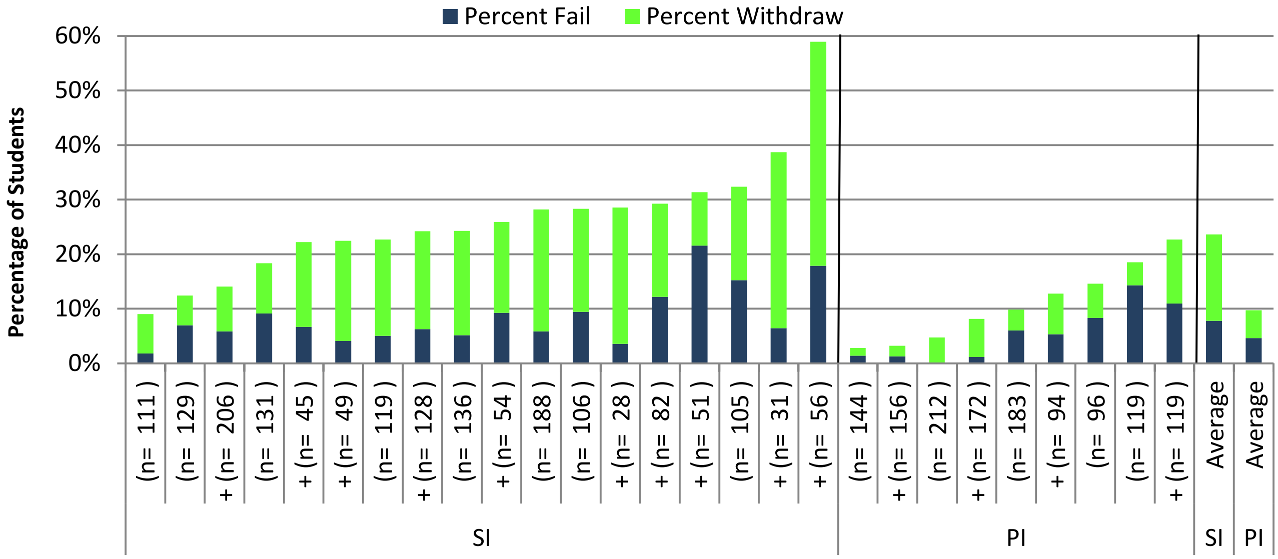

Figure 5. Longitudinal fail rates broken up by failures (DF) and withdraws (W) for each section of CS1. Sections are divided by SI or PI. The number of students (n) is provided. + denotes a class taught in the Winter or Spring.

6. DISCUSSION

6.1 Better Learning or Lowered Standards

In classes where instructors adopted PI, failure rates were, on average, reduced by more than half. Is this a good thing? Are more students learning what they need to pass the class or are the instructors lowering standards? This study does not allow us to answer these questions. We note that ultimate responsibility falls on the instructor to maintain the rigor of a course such that passing students have the requisite knowledge, understanding, and skills for future coursework and careers. The extent to which this is regularly measured may vary by institution depending on, for example, whether courses use common exams, have detailed course learning goals [17], and/or departments analyze exam scores or other student work.

"Making a course easier to pass" can be something to support—not avoid – depending on how it is accomplished. The obvious approach to avoid is to lower standards, effectively reducing the required knowledge, understanding, or skill required to pass. Alternatively, what if a course is "easier" without lowering standards? By designing a course to better support students in their attainment of learning goals, standards can be preserved while facilitating "easier" learning. For example, a course may be "easier" if is designed to offer students a better learning environment: one tailored to support learning experiences based on what is known about how the brain learns.

Whether PI facilitated such a learning environment or if standards eroded (or some combination of the two) during the ten years' worth of courses analyzed, we cannot say. We can say that none of our follow-on courses have seen a spike in failure rates, but that has issues in itself and provides only a partial answer. Even a post-hoc study would be of limited use as variation in course evaluation schemes may mean that for Course X an "A" could be earned with a 45% on the final exam whereas in Course Y 90% might be required for the same grade. A different study, in progress, that might inform these questions is to have a single instructor teach the "same" course with all the same components except for the teaching method [1].

We reiterate that we make an explicit assumption in this study that instructors did not intentionally or otherwise make their courses easier by reducing what students were expected to know. To the extent that this assumption may be violated, the results of the study may not be replicable.

Table 2. Statistical significance of failure rate comparisons.

Comparison	Fail Rate SI	Fail Rate PI	p-value
CS1	76.0%	90.2%	<0.01*
CS1.5	85.5%	89.2%	0.066
Theory	74.9%	94.2%	<0.01*
Architecture	84.1%	94.1%	<0.01*
Inst. 1	74.0%	84.5%	<0.01*
Inst. 2	82.4%	91.9%	<0.01*
Inst. 3	75.7%	96.3%	<0.01*
Inst. 4	77.4%	97.1%	<0.01*

6.2 CS 1.5

CS1.5 experienced a considerably lower change in fail rate. This can be partially explained by the low CS1.5 SI fail rate of 15%, as this provides less room for improvement. However, we suspect the more likely cause is that CS1.5 is the direct follow-on to CS1. In the SI classes, the high CS1 fail rate of 35% may have removed many of the students who might have struggled, contributing to the low SI CS1.5 fail rate. In the PI classes, the greater number of students who pass PI CS1 courses can continue on and pass PI CS1.5. Thus the finding of a large impact on the front-end and a smaller impact on the backend of this two course sequence is unsurprising.

6.3 Programmatic Responsibilities

These results can have a potentially large impact from a departmental or programmatic view in the following ways:

1. **Retention –** Students failing CS1 are at high risk for leaving the major. Having more students succeed in CS1 may result in higher retention of students in the major [14]. There is the potential that these additional students ultimately struggle in later classes, which we plan to study as our PI CS1 cohorts continue.
2. **Efficiency** – Particularly in upper-division courses needed for the major, students who fail the course are likely to repeat. The 16% reduction in failed students in our theory and architecture courses could potentially result in fewer (or smaller) courses needing to be taught.
3. **Time to Degree –** Student failures can result in additional terms before degree conferral. Although a more comprehensive study is warranted, our average time to degree is 4.3 years which is 0.3 years longer than the standard 4 year plan. Reducing failure rates by 60% could conceivably reduce time to degree to 4.1 years.
4. **Student Satisfaction** – Evaluating student failure as mere percentages can lose sight of the fact that failing a course can be devastating for students. Failure not only impacts self-worth but can also impact financial student aid and, in some cases, the ability of a student to stay at the institution. For those who persevere, one wonders at negative feelings they might harbor toward that experience. These feelings may result in lower degree/institution satisfaction and alumni participation.

The impact of these reductions in failure rates is an interesting topic within itself and is the topic of ongoing research. While it is undeniably the student's work (or lack of work) that fails a course, these results do contribute to a consideration of whether students would be right to wonder whether our instructional practices are supporting, or failing, them.

6.4 Recommendations

For readers interested in adopting PI in their courses to achieve similar reductions in fail rate, we provide the following recommendations:

1. **Best Practices** – Best Practices for Peer Instruction can be found online [13,16] and can serve as an excellent starting point for potential adopters.
2. **Materials** – The development of materials for PI requires time and effort and materials improve with use. We encourage potential adopters to review the materials available for a number of computer science classes online [13].
3. **Support –** Each of the instructors in this study, other than the initial adopter, had at least one PI mentor assisting them in the adoption of PI in their course. We highly recommend potential adopters seek out personal mentorship (feel free to contact the authors who can serve as mentors or point you to a potential mentor in your area). We are interested in further study of the factors supporting instructors in adoption of PI.

7. CONCLUSIONS

Classes implementing the Peer Instruction methodology have been shown to improve student learning outcomes in other STEM disciplines. This result is challenging to convincingly replicate in computing, due to our lack of standardized assessment instruments. However, we consider an important programmatic metric of success – student pass rates. Comparing against a 10-year, 10,000+ student, longitudinal set of course data from our own institution, we find that in courses where instructors adopt PI, course fail rates are reduced by an average of 61% (a reduction in absolute fail rates of 13%). Additionally, we identify that this effect is likely not due to student improvement with time or due to PI being implemented by "better" instructors. From a within-instructor comparison of four instructors, we find fail rates to decrease an average of 65% (a reduction in absolute fail rates of 15%). While this in-situ, retrospective study lacks some perhaps desirable controls, we believe its real-world nature lends the results additional credence.

8. REFERENCES

[1] Anonymous. Peer Instruction versus Lecture. Submitted for Publication. 2012

[2] Crouch, C. H., and Mazur, E. Peer instruction: Ten years of experience and results. *American J. of Physics 69*, 2001.

[3] Caldwell, J. E. Clickers in the large classroom: Current research and best-practice tips. *CBE-Life Sciences Education* 6, 2007.

[4] Cutts, Q., Esper, S., and Simon, B. Computing as the 4th "R": a general education approach to computing education. In *Pro. 7th ICER*. 2011.

[5] Eberlein T, Kampmeier J, Minderhout V, Moog RS, Platt T, Varma-Nelson P, White HB. Pedagogies of engagement in science. A comparison of PBL, POGIL, and PLTL. Biochem Mol Biol Educ 36, 262–273. 2008.

[6] Freeman S, O'Connor E, Parks JW, Cunningham M, Hurley D, et al. Prescribed active learning increases performance in introductory biology. *CBE-Life Scence Eduation. 6:132–39*. 2007.

[7] Knight, J. K., and Wood, W. B. Teaching more by lecturing less. *Cell Biology Education 4*, 2005.

[8] Lasry, N., Mazur, E., and Watkins, J. Peer instruction: From Harvard to the two-year college*," Am. J. Phys. 76 (11), 1066-1069*. 2008.

[9] Pargas, R. P., and Shah, D. M. Things are clicking in computer science courses. *Proc. of 37th SIGCSE*, 2006.

[10] Porter, L., Bailey-Lee, C., Simon, B., Cutts, Q., and Zingaro, D. Experience Report: A Multi-classroom Report on the Value of Peer Instruction. *Pro. of 16th ITiCSE*, 2011.

[11] Porter, L., Bailey-Lee, C. Simon, B., Zingaro, D. Peer Instruction: Do Students Really Learn from Peer Discussion*? In Proc. 7th ICER*, 2011.

[12] Prepare and Inspire: K–12 Education in Science, Technology, Engineering, and Math (STEM) for America's Future. Executive Office of the President, Washington, D.C., 2010.

[13] Simon, B. http://www.peerinstruction4cs.org/

[14] Simon, B., Kinnunen, P., Porter, L., Zazkis, D. Experience Report: CS1 for Majors with Media Computation. *In Proc. of 15th ITiCSE*. 2010.

[15] Simon, B., Kohanfars, M., Lee, J, Tamayo, K., and Cutts, Q. Experience report: Peer instruction in introductory computing. *Proc. of 41st SIGCSE,* 2010.

[16] UBC Science Education Initiatives. Clicker Resource Guide, http://cwsei.ubc.ca/resources/clickers.htm.

[17] UBC Science Education Initiatives. Learning Goals, http://cwsei.ubc.ca/resources/learn_goals.htm.

[18] Zingaro, D. Experience report: Peer instruction in remedial computer science. *Proc. of 22nd Ed-Media*, 2010.