ASEE 2022 ANNUAL CONFERENCE
Excellence Through Diversity
MINNEAPOLIS, MINNESOTA, JUNE 26TH-29TH, 2022 ASEE
Paper ID #37377

An Analytic Comparison of Student-Scheduled and Instructor-Scheduled Collaborative Learning in Online Contexts

Geoffrey L Herman (Teaching Associate Professor)

Dr. Geoffrey L. Herman is the Severns Teaching Associate Professor with the Department of Computer Science at the University of Illinois at Urbana-Champaign. He earned his Ph.D. in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign as a Mavis Future Faculty Fellow and conducted postdoctoral research with Ruth Streveler in the School of Engineering Education at Purdue University. His research interests include creating systems for sustainable improvement in engineering education, conceptual change and development in engineering students, and change in faculty beliefs about teaching and learning. He is an associate editor with the Journal of Engineering Education and a board member of the Computing Research Association Education committee.

Yucheng Jiang

Yueqi Jiang

Seth Poulsen (University of Illinois at Urbana - Champaign)

Matthew West (Prof.) (University of Illinois at Urbana - Champaign)

Mariana Silva

Mariana Silva is a Teaching Associate Professor in the Department of Computer Science at the University of Illinois at Urbana-Champaign. Silva is known for her teaching innovations and educational studies in large-scale assessments and collaborative learning. She has participated in two major overhauls of large courses in the College of Engineering: she played a key role in the re-structure of the three Mechanics courses in the Mechanical Science and Engineering Department, and the creation of the new computational-based linear algebra course, which was fully launched in Summer 2021. Silva research focuses on the use of web-tools for class collaborative activities, and on the development of online learning and assessment tools. Silva is passionate about teaching and improving the classroom experience for both students and instructors.

An Analytic Comparison of Student-Scheduled and Instructor-Scheduled Collaborative Learning in Online Contexts

Abstract

Collaborative learning can improve student learning, student persistence, and the classroom climate. While work has documented the tradeoffs of face-to-face collaboration and asynchronous, online learning, the trade-offs between asynchronous (student-scheduled) and synchronous (instructor-scheduled) collaborative and online learning have not been explored. Structured roles can maximize the effectiveness of collaborative learning by helping all students participate, but structured roles have not been studied in online settings. We performed a quasi-experimental study in two courses—Computer Architecture and Numerical Methods—to compare the effects of asynchronous collaborative learning without structured roles to synchronous collaborative learning with structured roles. We use a data-analytics approach to examine how these approaches affected the student learning experience during formative collaborative learning assessments. Teams in the synchronous offering made higher scoring submissions (5-10% points better on average), finished assessments more efficiently (11-16 minutes faster on average), and had greater equality in the total number of submissions each student made (for example, significant increase of 13% in the mean equality score among all groups).

Introduction and Background

Collaborative learning can improve student persistence, learning outcomes, and classroom cultures [1, 2]. Successful and productive collaborations are not guaranteed. Collaboration can be greatly improved by careful design of learning tasks [3, 4], assignment of team roles [5], and the use of technologies [6, 7].

Many evidence-based practices for collaborative learning, such as Context-Rich Collaborative Problem Solving [8] or Process-Oriented Guided Inquiry Learning [9], were developed for in-person, synchronous learning contexts. With the on-going pandemic, the importance of online only pedagogies has become more readily apparent. Online pedagogies provide new opportunities for increased access to evidence-based pedagogies at potentially lower cost and greater ability to scale. Unfortunately, we do not know much about how to effectively provide synchronous collaborative learning experiences only online.

As instructors at a residential campus that moved to fully remote instruction because of the pandemic, we wanted to offer the best collaborative learning experience we could so that our students could still experience some of the benefits of a residential campus: connecting with other amazing students. So, we wanted to particularly explore how to support synchronous online, collaborative learning that maximized students' direct interactions with each other. Without much direct, clear guidance from the research literature about best practices, most research on online collaboration focuses on purely asynchronous learning, we sought to study the effect of different policies on the student learning experience during online, synchronous collaborative learning.

We compared a more student-centric model where students determined when they would meet as a group (student-scheduled synchronous) and how their group would function (free-for-all roles) with a more instructor-centric model where the instructor determined when students would meet as a group (instructor-scheduled synchronous) and provided instructions and constraints for how group members interacted (structured roles).

To inform the design of our research study, we discuss two critical parameters for supporting productive collaborations: synchronous vs. asynchronous modalities [3, 10, 11] and structured roles [12, 9, 13].

Synchronous vs asynchronous learning

Collaborative-learning can be delivered in face-to-face environments and online through the Internet [14, 15, 16, 17, 18]. Participants can contribute to the task at different times (asynchronous activity) or interact simultaneously (synchronous activity). Most evidence-based practices to maximize collaborative learning have relied on the assumption of face-to-face interactions [19, 20] or online asynchronous interactions, where group members communicate via chat message systems, online forums and e-mails [21, 22]. Little research has explored the use of synchronous, collaborative learning in online contexts, but some work suggests that online tools for synchronous collaborative learning may promote more equal participation among students than strictly face-to-face learning [11].

Synchronous and asynchronous modalities provide different benefits for collaborative learning. Asynchronous classes can increase access and equity, enabling students from differing time zones, students with work schedules, or students with limited access to computing resources the opportunity to participate when they are able [23]. Additionally, asynchronous classes can better support students with disabilities or language barriers [24]. In contrast, synchronous learning opportunities are associated with greater engagement among the students who can attend the synchronous learning activities. Our study seeks to explore these trade-offs from a data analytics perspective.

Structured roles

A lack of clarity and experience in assuming team roles can lead students to default into domineering team leaders or passive free-loaders [25]. Evidence-based practices such as pair programming [12], role scripting [26, 27] and Process Oriented Guided Inquiry Learning (POGIL) [9, 13] have shown that providing students with structured roles can help them participate more equally during collaborative learning. Structured roles are designed to create positive interdependence between the roles.

In our classes, we based our structured roles on POGIL roles. The "recorder" writes the team's answers to problems, the "manager" is responsible for keeping the team on task, and the "reflector" is responsible for guiding the team in reflection activities on their learning process. POGIL has primarily been implemented only in-person contexts. The COVID-19 pandemic has led to some informal sharing of best practices among POGIL instructors [28, 29], but there have not been formal studies of how to best adapt POGIL roles to online learning contexts.

Research questions

We studied the implementation of online, collaborative learning in two large-enrollment, required computing courses: Computer Architecture and Numerical Methods. Both courses were offered during the Fall and Spring semesters of the 2020-21 academic school year and were offered only online due to the COVID-19 pandemic. In Fall 2020, instructors for both courses agreed to primarily support student-scheduled, synchronous, collaborative learning in an effort to best accommodate students across a variety of time zones and students from lower socio-economic backgrounds who might not be able to easily attend instructor-scheduled, synchronous class sessions [24]. In Spring 2021, both courses switched to an instructor-scheduled synchronous model of instruction in an effort to create stronger senses of community among students and to better support students' socio-emotional needs during the pandemic [30, 31, 32]. In tandem, both instructors shifted from letting students collaborate in whatever manner they saw best (free-for-all roles) in Fall 2020 to requiring students to take on POGIL-inspired structured roles (recorder, manager, and reflector) in Spring 2021.

We conducted a quasi-experimental study to compare these policies and explore two research questions.

RQ1: What effect do student-scheduled, synchronous classes with free-for-all roles and instructor-scheduled synchronous classes with structured roles have on the student learning experience during collaborative learning activities? We want to encourage students to help each other learn and complete the learning exercises. We hypothesize that if students are indeed helping each other collaboratively, then they will derive higher-scoring solutions to problems and complete collaborative exercises more efficiently.

RQ2: What effect do student-scheduled synchronous classes with free-for-all roles and instructor-scheduled synchronous classes with structured roles have on the equality of the number of students' contributions during collaborative learning activities? We want all students to actively participate during collaborative learning activities. We don't want students to become disenfranchised and become freeloaders and we don't want students to take over and dominate group efforts. We explore this question by comparing the equality of the number of submissions that students make.

Research methods

We performed two trials of a quasi-experimental study in Computer Architecture and Numerical Methods. Computer Architecture and Numerical Methods are required for computer science majors at a public, research-intensive university. The courses each enroll 300–400 students per semester. They meet twice per week with Computer Architecture convening 2 hours per meeting and Numerical Methods convening 1.5 hours per meeting.

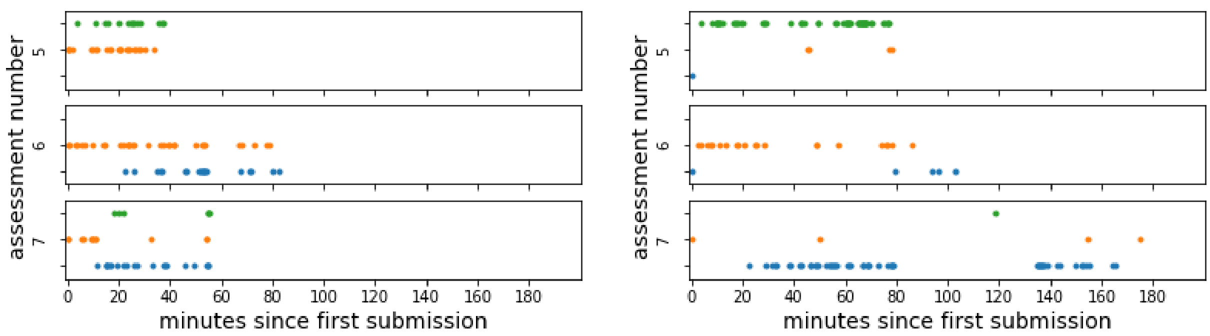
Both courses adopt a flipped classroom format and deliver all assessments (including pre-lecture content with short checkpoints, homework, longer form machine problems, frequent quizzes and in-class collaborative learning assessments) through the open-source online assessment platform PrairieLearn [33]. We compare click-stream data from PrairieLearn for the collaborative learning assessments (called **Group Assessments, henceforth GAs**) to observe how different policies (student-scheduled synchronous with free-for-all roles versus instructor-scheduled synchronous with structured roles) affected how students collaborated during these assessments. For brevity, we will henceforth call student-scheduled synchronous as "asynchronous." All aspects of the courses were kept the same between semesters: same instructors, same learning materials, same homework/quiz structures, and same course policies aside from changes related to GAs.

PrairieLearn is a problem-driven learning environment designed to promote mastery-based learning, where students are able to practice solving randomized problem variants repeatedly, receiving immediate feedback about their current mastery level. It also provides autograding for a wide range of question types including numerical and symbolic questions, programming problems, and some computer-assisted drawing tasks. When used for GAs, all members of the same team share the same assessment and therefore the corresponding grade. During a class session, students work collaboratively to solve each question, with the ability to submit unlimited answers, until they get the question marked as correct. They receive both feedback from PrairieLearn and from course staff that are available during class time.

In Computer Architecture, GAs had a mixture of problem types, including comparative analysis of real-world systems, interactive design activities, and short programming problems. Most GAs had between 4 and 10 problems. In Numerical Methods, GAs mostly consisted of a multi-part programming problem with students encouraged to submit code incrementally.

Student-scheduled synchronous ("asynchronous"), free-for-all roles course policies. For "asynchronous" offerings of the courses, instructors did not require attendance during class time but provided office hours in Zoom during scheduled times. At the start of the semester, students completed a survey to indicate approximate times that they would like to complete the GAs and a classmate they would like to have assigned to their team. The instructors assigned students to teams of 3–4 based on their time preferences and classmate preferences. Students were expected to complete GAs together during a mutually agreed upon time. Lecture material to help students complete GAs were pre-recorded and available to students asynchronously.

During GAs, students were allowed to complete them with whatever roles they thought was best. In Figure 1a, we visualize an example click-stream log from three GAs from the asynchronous term. Each numbered rectangle represents the click-stream data from a different GA, and the x-axis represents the first 180 minutes that any member of the team worked on a GA. Each dot represents a submission by a team member and each member of the team is shown consistently on a different row. These visualizations show that students often made submissions during the same time intervals, indicating that students were working at the same time as hoped for. Some students did not submit anything on some GAs. These lack of submissions may or may not indicate



(a) Asynchronous course offering

(b) Synchronous course offering

Figure 1: Visualization of click-stream data. Each subplot represents a different assessment (5,6,7), each dot inside a subplot represents a student submission into PrairieLearn, and each color represents a unique student. For better clarity, we display different students in separate rows inside the subplot. (a) Submissions for a selected group in the asynchronous course offering. The submissions happen concurrently, indicating that students are working together. (b)Submissions for a selected group in the synchronous course offering. One student makes the majority of submissions in each assessment, indicating that the student was likely assigned to the role of recorder for the team.

whether students were present and collaborating with their team members, as a team member could be helpfully giving feedback while not actively pressing a submit button.

Instructor-scheduled synchronous, structure roles course policies. For synchronous offerings, instructors required attendance during class time. Both classes offered an additional section to accommodate students in time zones that did not align with the scheduled course time (e.g., a 5 a.m. section for students in a different hemisphere). Attendance was collected by using Zoom's attendance record. For the first two weeks, students were assigned into random ad hoc teams to give them opportunities to meet other students. Starting in the third week, students were allowed to self-organize into teams of 3–4 and students who did not self-organize were assigned to teams of 3–4 by the instructor. The teams remained fixed for the following 6 weeks of the semester. For the last 6 weeks, students were given the opportunity to remain with their assigned teams or to choose new teams. Most teams (>90%) elected to stay with their originally assigned teams.

At the start of the semester, students were taught about structured roles and their potential benefits for team dynamics. During GAs, students first had to complete a "manager report" where the manager indicated which team members were in attendance and what roles each member was assigned to. The recorder was then responsible for submitting all remaining answers for the team into PrairieLearn. The reflector completed a "reflector survey" at the end of the assessment to reflect on what the team had learned, the team's preparation for the assessment, and how the team had worked together. To receive participation points, students were required to participate in each role twice during the semester and were encouraged to rotate their roles each class meeting.

In Figure 1b, we show another visualization of click-stream data that demonstrates how one student made the vast majority of submissions on each GA, indicating they were likely the recorder. Each GA has a different student making the majority of submissions, indicating the teams were rotating which student was the recorder.

Data collection

At the beginning of each term, students provided consent for their anonymized data to be used as part of research studies. PrairieLearn collects all click-stream data: a log of every time a student views a question, grades a question, views an assessment. We downloaded all click-stream logs for GAs from both Computer Architecture and Numerical Methods after all final grades had been assigned to ensure that log data did not influence how the instructors taught their students.

Data cleaning

To account for students dropping a course, team compositions being shuffled, and students who simply refused to work with their teams, we analyzed only those teams where at least two students worked together for more than half of GAs in a semester. Because students were allowed to drop their lowest GA scores or they simply failed to complete every GA, we report the number of teams that completed each GA and analyze statistics based only on the teams that completed each GA.

Metrics for Research Question 1

We defined two metrics to describe the quality of the student learning experience during GAs: the performance of submissions and the time to completion. We hypothesize that if group members are helping each other learn, they should make higher performing submissions and they would spend less time to finish an assignment. Analyzing both metrics is important for observing productive collaboration. For example, a team might reduce their time to completion by using a divide-and-conquer rather than collaborative approach, but because team members are not actively helping each other, we would not expect to also see a corresponding improvement in the performance of submissions.

Performance of Submissions: when using PrairieLearn, students can make unlimited submission to the same question without being penalized. Therefore, we defined the performance of submissions as the team's average submission score made during an assessment. If a team had a better understanding of the course material, we expect they would achieve higher average submission scores. For example a team that earned 90% on their first submission and 100% on their final submission would have an average score of 95% while a team that earned 0% on their first submission, 60% on their second submission and 100% on their final submission would have an average score of 53.3%. We would estimate that the first team had a better understanding of the course material or the first team at least proofread their submissions better. Because students were allowed to resubmit answers without penalty until they got them marked as correct, we cannot use the final assessment score as a performance metric, because most teams eventually earned perfect scores.

Time to Completion: We summed the time between submissions, removing submissions made more than 60 minutes apart. We assumed this amount of time indicated that a team was idle and not actively working on the assessment during that time interval.

Metrics for Research Question 2

When implementing collaborative learning, we want to minimize the number of freeloaders or dominators and encourage all team members to actively engage in the learning process for the whole semester. Because we expect that one member, the recorder, should make the majority of submissions in each GA during the synchronous offering, but had no such expectation for the "asynchronous" offering, we evaluated team contributions on a per semester basis rather than a per assessment basis to make the comparison fair.

Figure 2 illustrates the percentage of submissions for each team member from three different teams. To define a standard metric for equality, we computed the standard deviation of the percentage of submissions for each team member (σ_p). Because the group size varied between 2–4 and group size affects the maximum possible standard deviation ($\sigma_{\max}$), we normalized our metric by dividing the standard deviation by the maximum possible standard deviation. To improve interpretability, we subtract this quantity from 1 to get the **equality score** e, giving

$$e = 1 - \frac{\sigma_p}{\sigma_{\max}}. \tag{1}$$

Teams with perfectly equal number of submissions from all team members would yield an equality score of 1 and a team where one team member makes all submissions would yield an equality score of 0. In Fig. 2, group 19 had an equality score of 0.46 because student 2 made most of the submissions and student 3 made almost no submissions. Group 94 had an equality score of 0.94 because team members made roughly the same number of submissions.

Analysis methods

For our first research question, we performed a meta-analysis across all GAs for each course and each metric (performance of submissions and time to completion) using a Multi-Level Modeling (MLM) approach [34]. We chose MLM because it can model nested data (i.e., assignments nested within a course) with different variances across groups [34, 35, 36] as is the case with our data. We use the following MLM for performance of groups on GAs:

$$\text{Outcome}_{ij} = \beta_0 + \beta_1 \text{Sync} + W_j + \epsilon_{ij}, \tag{2a}$$
$$W_j \sim N(0, \tau^2), \tag{2b}$$
$$\epsilon_{ij} \sim N(0, \sigma^2), \tag{2c}$$

where i indexes groups of students and j indexes GAs. Outcome$_{ij}$ represents the outcome (either performance or duration, we will use the same model for both) of group i on GA j. Sync represents whether assignment j was done synchronously or asynchronously (1=synchronously, 0=asynchronously). Here β_0 is the intercept, which in this model is the average outcome for groups doing the assignment asynchronously. In our case, the parameter we are most interested in is β_1, which estimates the impact of doing an assignment synchronously. The error term at the GA level is ϵ_{ij}, which models the variance in the distribution of the performance of different

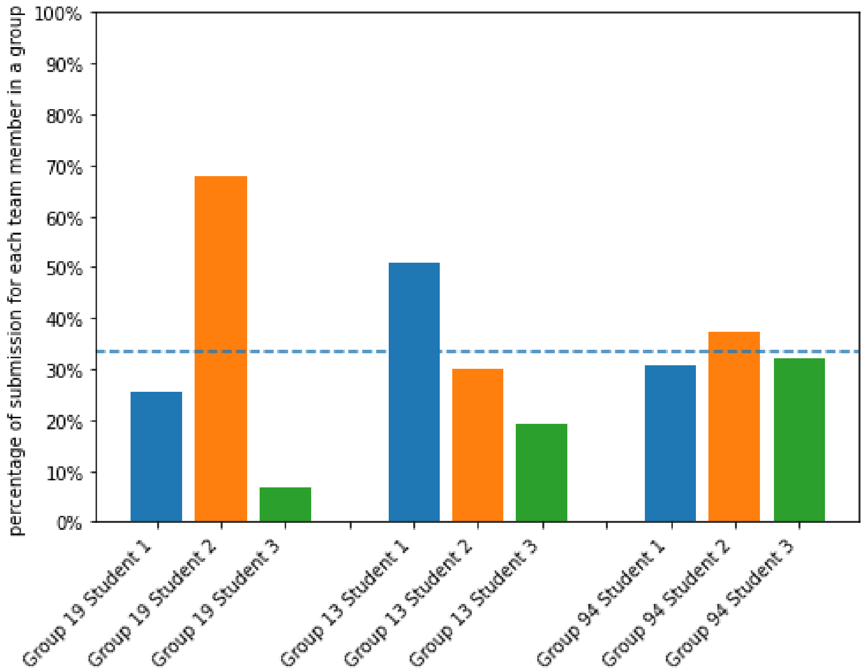


Figure 2: Histograms for three teams with varying levels of equality in their percentage of submissions: Group 19 Equality Score = 0.46, Group 13 Equality Score = 0.72, Group 94 Equality Score = 0.94

groups on a particular GA as a normal distribution with variance σ^2. The course-level error term is W_j, which models the variance in outcomes between different GAs as a normal distribution with variance τ^2. Because the variation between performance on the same assignment is the focus here, as our standardized effect size, we report the value of β_1 divided by the standard deviation of the GA-level error term (sd(σ)). We fit this model four times—once time for each outcome (performance and duration) for each course (Numerical Methods and Computer Architecture). We fit our MLM using the lme4 library in the R programming language [37, 38].

The outcomes over all groups and all assessments vary over some range. We used the intra-class correlation coefficient (ICC) to investigate whether this variance of outcomes is a result from variation **between assessments** (for example, some assessments may be harder than others, but groups have similar outcomes within the same assessment) or variation **within assessments** (where groups have different outcomes on the same assessment). The ICC is defined as the ratio between the variance between assessments and the total variance:

$$\mathrm{ICC} = \frac{\tau^2}{\tau^2 + \sigma^2}. \tag{3}$$

If the ICC is near 0, then almost all the variation is between assessments. If the ICC is near 1, then almost all the variation is within assessments.

To aid interpretation of the MLM, we compare averages across the terms using two-way t-tests with $\alpha = 0.05$ for each assessment individually. We report effect sizes using Hedge's g because of the unequal sample sizes. We do not apply a correction to the α levels because these statistical tests are merely illustrative rather than the main statistical tests for our study.

For our second research question, we compared the average equality score between semesters using a two-way t-test again using $\alpha = 0.05$. We performed this analysis for each course individually and then aggregated our findings across both courses by combining the data from both courses from the same term and again compared the means using another independent samples, two-way t-test. We report effect sizes using Hedge's g because of the unequal sample sizes. We again view only the aggregated statistic as the primary measure and use the disaggregated data to aid interpretation.

Results

We report the result of the MLMs for each course and for performance of submissions and time to completion. We provide statistical comparisons for each GA individually to aid interpretation.

Performance of submissions

Table 1 shows the summary of the statistics for all the GAs in the Computer Architecture course. The synchronous offering had significantly higher performing submissions than the "asynchronous" offering on 14 of 22 assessments. In only 2 assessments the performance of submissions from the asynchronous offering were better, but the results were not significant.

Table 1: Performance of submissions and time to complete for Computer Architecture for each GA across "asynchronous" (FA20) and synchronous (SP21) offerings. Performance of submissions is measured by finding the average score of every submission by each team (instead of percent, we display scores between 0-1), time interval counts if two submissions are made within 60 minutes and n_{FA20} and n_{SP21} are the number of submissions in each semester. The (*) denotes the results that are statistically significant.

			Performance of submissions				**Time to complete**			
GA	n_{FA20}	n_{SP21}	FA20 score(std)	SP21 score(std)	p-value	Hedge's g	FA20 duration(std)	SP21 duration(std)	p-value	Hedge's g
1	113	112	.79(0.1)	**.82(0.11)**	0.01*	0.36	82.97(58.58)	**59.86(31.66)**	<0.01*	0.49
2	105	113	.75(0.11)	.73(0.09)	0.36	-0.13	62.06(41.14)	59.72(15.95)	0.59	0.08
3	109	112	.47(0.14)	**.57(0.15)**	<0.01*	0.67	99.56(48.28)	**77.12(27.61)**	<0.01*	0.57
4	110	113	.55(0.17)	.56(0.16)	0.52	0.09	62.1(36.79)	54.85(20.42)	0.07	0.24
5	110	113	.36(0.17)	**.41(0.16)**	0.02*	0.32	92.95(49.7)	**75.57(30.76)**	<0.01*	0.42
6	109	113	.75(0.14)	.77(0.14)	0.24	0.16	136.23(74.44)	116.37(90.46)	0.08	0.24
7	109	112	.79(0.15)	**.86(0.12)**	<0.01*	0.51	100.9(74.12)	**70.49(41.92)**	<0.01*	0.51
8	108	112	.82(0.21)	**.93(0.07)**	<0.01*	0.73	71.34(64.0)	**48.79(25.61)**	<0.01*	0.46
9	109	112	.78(0.1)	**.82(0.08)**	<0.01*	0.42	**18.99(16.74)**	30.46(14.58)	<0.01*	-0.73
10	109	111	.77(0.12)	.79(0.13)	0.19	0.18	38.8(22.97)	**30.48(11.62)**	<0.01*	0.46
11	107	111	.77(0.11)	**.82(0.12)**	0.01*	0.34	58.04(37.09)	**47.57(17.39)**	0.01*	0.36
12	107	112	.66(0.11)	**.70(0.09)**	0.02*	0.32	80.6(42.14)	**67.24(26.91)**	0.01*	0.38
13	106	106	.59(0.12)	.60(0.12)	0.71	0.05	84.65(54.73)	73.67(39.27)	0.09	0.23
14	106	112	.76(0.17)	.80(0.16)	0.07	0.25	44.87(25.13)	**35.88(16.75)**	<0.01*	0.42
15	104	110	.76(0.14)	.74(0.17)	0.26	-0.15	46.23(25.83)	43.35(19.03)	0.36	0.13
16	105	108	.43(0.24)	**.55(0.25)**	<0.01*	0.48	70.3(49.52)	62.28(36.04)	0.18	0.18
17	105	108	.50(0.17)	**.60(0.16)**	<0.01*	0.62	89.34(51.28)	81.54(36.58)	0.2	0.17
18	104	110	.54(0.16)	.57(0.17)	0.15	0.2	94.59(56.99)	**77.97(46.42)**	0.02*	0.32
19	105	107	.69(0.14)	**.74(0.13)**	0.02*	0.33	65.97(45.04)	56.09(24.79)	0.05	0.27
20	103	109	.50(0.19)	**.62(0.21)**	<0.01*	0.57	44.24(33.16)	38.74(27.98)	0.2	0.18
21	105	109	.79(0.07)	**.82(0.07)**	0.01*	0.39	27.55(17.78)	28.74(14.98)	0.6	-0.07
22	103	106	.63(0.18)	**.71(0.19)**	<0.01*	0.41	30.27(27.4)	26.88(26.51)	0.36	0.13

Table 2 shows the MLM model fit when Outcome$_{ij}$ is the average submission score per team i on GA j in the Computer Architecture class. The submissions by synchronous groups were, on average, 4.86 percentage points better than submissions by "asynchronous" groups. The ICC for this model was 0.44, meaning that just over half of the unexplained variance in the model came from the variance between assessments, and the rest came from the variance between groups working on the same assessment. The standardized effect size ($\beta_1/\text{sd}(\sigma)$) was small (0.37) but significant ($p < 0.001$).

Table 2: MLM model fit for the average submission data in the Computer Architecture class, showing that groups in the synchronous course performed 4.86 percentage points better on average.

	Estimate (std)	t-value		Estimate (std)
β_0 (Intercept)	65.76 (2.84)	23.12	σ^2	175.85 (13.26)
β_1 (Sync)	4.86 (0.43)	11.23	τ^2	224.06 (14.97)

Table 3 shows the summary of the statistics for all the GAs in the Numerical Methods course. The synchronous offering had significantly higher performing submissions than the "asynchronous" offering on 8 of 9 assessments. In only 1 assessment the performance of submission from the asynchronous offering was better, but the result was not significant.

Table 4 shows the MLM model fit when Outcome$_{ij}$ is the average submission score per team i on GA j in the Numerical Analysis class. The submissions by synchronous groups were, on average, 9.63 percentage points better than submissions by "asynchronous" groups. The ICC for this model was 0.22, meaning that most of the unexplained variance in the model came from the variance between assessments, and the rest came from the variance between groups working on the same assessments. The standardized effect size ($\beta_1/\text{sd}(\sigma)$) is large (1.31) and significant ($p < 0.001$).

Table 3: Performance of submissions and time to complete for Numerical Methods for each GA across "asynchronous" (FA20) and synchronous (SP21) offerings. Performance of submissions is measured by finding the average score of every submission by each team (instead of percent, we display scores between 0-1), time interval counts if two submissions are made within 60 minutes and n_{FA20} and n_{SP21} are the number of submissions in each semester. The (*) denotes the results that are statistically significant.

			Performance of submissions				**Time to complete**			
GA	n_{FA20}	n_{SP21}	FA20 score(std)	SP21 score(std)	p-value	Hedge's g	FA20 duration(std)	SP21 duration(std)	p-value	Hedge's g
1	121	93	.34(0.13)	**.54(0.2)**	<0.01*	1.23	95.22(50.34)	**55.14(24.42)**	<0.01*	0.97
2	121	95	.46(0.16)	**.59(0.17)**	<0.01*	0.75	59.13(35.91)	**46.63(20.35)**	<0.01*	0.41
3	119	95	.38(0.09)	**.46(0.13)**	<0.01*	0.77	72.05(36.37)	68.88(39.0)	0.54	0.08
4	118	92	.58(0.15)	**.64(0.11)**	<0.01*	0.5	98.33(50.48)	**58.42(26.86)**	<0.01*	0.95
5	119	92	.40(0.14)	**.46(0.09)**	<0.01*	0.57	82.41(55.45)	**67.11(26.94)**	<0.01*	0.34
6	118	90	.55(0.15)	**.61(0.13)**	<0.01*	0.4	51.86(30.71)	58.63(25.22)	0.08	-0.24
7	64	90	.51(0.12)	**.66(0.13)**	<0.01*	1.25	58.92(43.9)	48.44(20.98)	0.08	0.32
8	116	43	.46(0.1)	**.62(0.14)**	<0.01*	1.42	49.8(22.5)	61.76(39.86)	0.07	-0.42
9	116	85	.56(0.11)	.54(0.14)	0.26	-0.17	81.41(39.0)	**50.49(28.44)**	<0.01*	0.88

Time to completion

As described in Table 1, the synchronous offering of Computer Architecture spent significantly less time than the "asynchronous" offering to complete 10 of 22 assessments and significantly

Table 4: MLM model fit for the average submission data in the Numerical Methods class showing that groups in the synchronous course performed 9.63 percentage points better on average.

	Estimate (std)	t-value		Estimate (std)
β_0 (Intercept)	47.18 (2.48)	19	σ^2	53.78 (7.33)
β_1 (Sync)	9.63 (0.67)	14.46	τ^2	191.12 (13.82)

more on 1 assessment. Table 5 shows the MLM model fit when Outcome$_{ij}$ is the time to complete GA j for team j in the Computer Architecture class. The synchronous groups completed assessments 10.93 minutes faster on average than "asynchronous" groups. The ICC for this model was 0.27, meaning that most of the unexplained variance in the model came from the variance between assessments, and the rest came from the variance between groups working on the same assessment. For this model, the standardized effect size ($\beta_1/\text{sd}(\sigma)$) is medium sized ($-0.43$) and significant ($p < 0.001$), meaning that students in the synchronous class spent less time on the assessments than "asynchronous students". Summing across all assessments, the medium-effect size translates to synchronous students in Computer Architecture spending 3.98 hours less or 1 full week of instructional time on in-class activities.

Table 5: MLM model fit for the time to complete data in the Computer Architecture class, showing that groups in the synchronous course completed their work 10.93 minutes faster on average.

	Estimate (std)	t-value		Estimate (std)
β_0 (Intercept)	68.34 (5.37)	12.72	σ^2	620.14 (24.9)
β_1 (Sync)	-10.93 (1.17)	-9.31	τ^2	1646.09 (40.57)

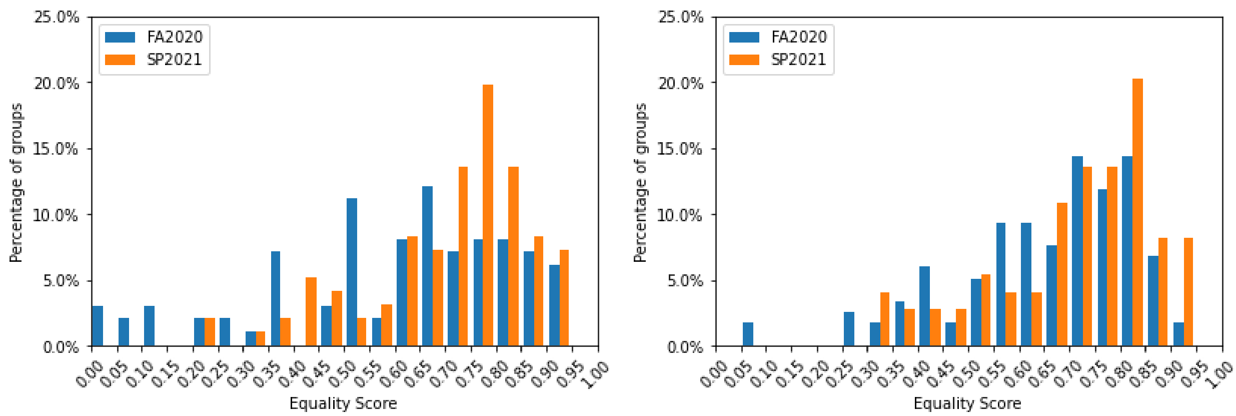
As described in Table 3, the synchronous offering of Numerical Methods spent significantly less time than the "asynchronous" offering to complete 5 of 9 assessments. Table 6 shows the MLM model fit when Outcome$_{ij}$ is the time to complete GA j for team j in the Numerical Methods class. The synchronous groups completed assessments 16.19 minutes faster on average than "asynchronous" groups. The ICC for this model was 0.087, meaning that almost all of the unexplained variance in the model came from the variance between assessments, and most groups completed each assessments in a similar amount of time. For this model, the standardized effect size ($\beta_1/\text{sd}(\sigma)$) is large ($-1.41$) and significant ($p < 0.001$), meaning the synchronous students spent less time on assessments than "asynchronous" students. Summing across all assessments, the large effect size translates to synchronous students in Numerical Methods spending 2.83 hours of instructional time less than students in the "asynchronous" offering (almost 1 full week) on in-class activities.

Table 6: MLM model fit for the time to complete data in the Numerical Methods class, showing that groups in the synchronous course completed their work 16.19 minutes faster on average.

	Estimate (std)	t-value		Estimate (std)
β_0 (Intercept)	72.35 (4.01)	18.04	σ^2	132.28 (11.5)
β_1 (Sync)	-16.19 (1.79)	-9.02	τ^2	1390.96 (37.3)

Equality of number of submissions

Figure 3a provides a histogram of the equality score for all teams with more than 2 members from Computer Architecture. The synchronous offering of the course ($\mu = 0.71$, $\sigma = 0.17$, $N = 96$) had significantly more ($p = 0.006$, $g = 0.397$) equality in the number of submissions among team members than the "asynchronous" offering of the course ($\mu = 0.63$, $\sigma = 0.25$, $N = 99$).



(a) Computer Architecture

(b) Numerical Methods

Figure 3: Histogram of the equality scores for each team for the "asynchronous" (FA20) and synchronous (SP21) offerings.

Figure 3b provides a histogram of the equality score for all teams with more than 2 members from Numerical Methods. The synchronous offering of the course ($\mu = 0.71$, $\sigma = 0.16$, $N = 74$) had significantly more ($p = 0.049$, $g = 0.284$) equality in the number of submissions among team members than the "asynchronous" offering of the course ($\mu = 0.67$, $\sigma = 0.18$, $N = 118$).

Combining data reveals that the synchronous offering of the courses yielded significantly higher ($p < 0.001$) equality in the number of student submissions with a small effect size ($g = 0.337$).

Discussion and Conclusions

Our analysis reveals marked improvements across all metrics for synchronous collaborative learning with structured roles relative to "asynchronous" collaborative learning without structured roles. The effect sizes were substantive, suggesting improvements in the performance of submissions by a half to a full letter grade and students needing one week less of in-class time to finish assessments. These benefits were complemented by evidence that there was more equal participation by all team members in the synchronous offering.

Our data analytics approach cannot capture many nuances of collaborative learning that do not require clicking a button in PrairieLearn such as conversations or help seeking. Thus, we might see improvements in our chosen metrics that do not correspond to improved learning experiences. Counter-productive divide-and-conquer approaches might improve time to duration statistics or a dominant student might lead to improved performance of submissions. However, divide-and-conquer approaches likely won't also lead to improved submission performance as students are not checking each other's work and a dominant student would likely not lead to an equal number of submissions. When taken together, the improvements in all three metrics provide strong evidence that the synchronous course structured roles created an environment where students more readily participated in learning activities and helped each other collaboratively. In future papers, we will be complementing this analysis with deeper analysis of more qualitative data including students' feedback on reflector surveys, end-of-semester surveys, and observations of student teams during class time.

It is not clear whether the change from "asynchronous" to synchronous or free-for-all roles to structured roles or the combination of changes is responsible for the improvements in students' in-class experience. The instructors of these courses did find that the quality of their interactions with students and the logistics of running the course were also dramatically improved. For example, the synchronous meeting time made it easier for instructors to remind students of roles and provide tips and suggestions in real time for how specific teams could enact their roles. Based on our experience as instructors and the evidence from the data analytics, it is reasonable to claim that the combination of synchronous collaborative learning with structured roles is a best practice to recommend to other instructors.

References

[1] S. Freeman, S. Eddy, M. McDonough, M. Smith, N. Okoroafor, H. Jordt, and M. Wenderoth, "Active learning increases student performance in science, engineering, and mathematics," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 23, pp. 8410–8415, 2014.

[2] J. Gasiewski, M. Eagan, G. Garcia, S. Hurtado, and M. Chang, "From gatekeeping to engagement: A multicontextual, mixed method study of student academic engagement in introductory stem courses," *Research in Higher Education*, vol. 53, p. 229–261, 2012.

[3] M. Kapur and C. K. Kinzer, "Examining the effect of problem type in a synchronous computer-supported collaborative learning (cscl) environment," *Educational Technology Research and Development*, vol. 55, pp. 439–459, 2007.

[4] T. Tucker, S. Shehab, E. Mercier, and M. Silva, "Board 50: Wip: Evidence-based analysis of the design of collaborative problemsolving engineering tasks," *Proceedings of American Society for Engineering Education*, 2019.

[5] H. H. Hu, C. Kussmaul, B. Knaeble, C. Mayfield, and A. Yadav, "Results from a survey of faculty adoption of process oriented guided inquiry learning (pogil) in computer science," in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, 2016, pp. 186–191.

[6] T. Nokes-Malach, J. Richey, and S. Gadgil, "When is it better to learn together? insights from research on collaborative learning," *Educational Psychology Review*, vol. 27, p. 645–656, 2015.

[7] S. Shehab, L. Lawrence, E. Mercier, A. Margotta, E.Livingston, M. Silva, and T. Tucker, "Towards the effective implementation of collaborative problem solving in undergraduate engineering classrooms: co-designing guidelines for teaching assistants," *Proceedings of American Society for Engineering Education*, 2020.

[8] P. Heller and M. Hollabaugh, "Teaching problem solving through cooperative grouping. part 2: Designing problems and structuring groups," *American Journal of Physics*, vol. 60, no. 7, pp. 637–644, 1992. [Online]. Available: https://doi.org/10.1119/1.17118

[9] “Pogil: Process oriented guided inquiry learning,” https://pogil.org, Last accessed on 2021-01-31.

[10] M. H. Dlab, I. Boticki, N. Hoic-Bozic, and C. K. Looi, “Exploring group interactions in synchronous mobile computer-supported learning activities,” *Computers & Education*, vol. 146, p. 103735, 2020.

[11] R. R. Fowler, “Talking teams: Increased equity in participation in online compared to face-to-face team discussions,” *The ASEE Computers in Education (CoED) Journal*, vol. 6, no. 1, p. 14, 2015.

[12] B. Hanks, S. Fitzgerald, R. McCauley, L. Murphy, and C. Zander, “Pair programming in education: a literature review,” *Computer Science Education*, vol. 21, no. 2, pp. 135–173, 2011.

[13] R. S. Moog and J. N. Spencer, *Process oriented guided inquiry learning*. American Chemical Society Washington, DC, 2008, vol. 994.

[14] T. Koschmann, “Paradigm shifts and instructional technology: An introduction,” *CSCL: Theory and practice of an emerging paradigm*, vol. 12, no. 4, pp. 18–19, 1996.

[15] E. Lehtinen, K. Hakkarainen, L. Lipponen, M. Rahikainen, and H. Muukkonen, “Computer supported collaborative learning: A review,” *The JHGI Giesbers reports on education*, vol. 10, p. 1999, 1999.

[16] L. Lipponen, “Exploring foundations for computer-supported collaborative learning.” in *CSCL*, vol. 2, 2002, pp. 72–81.

[17] P. Dillenbourg, S. Järvelä, and F. Fischer, “The evolution of research on computer-supported collaborative learning,” in *Technology-enhanced learning*. Springer, 2009, pp. 3–19.

[18] P. Dillenbourg, *Collaborative learning: Cognitive and computational approaches. advances in learning and instruction series.* ERIC, 1999.

[19] E. Mercier and S. Higgins, “Collaborative learning with multi-touch technology: Developing adaptive expertise,” *Learning and Instruction*, vol. 25, p. 13–23, 2013.

[20] L. Paquette, N. Bosch, E. Mercier, J. Jung, S. Shehab, and Y. Tong, “Matching data-driven models of group interactions to video analysis of collaborative problem solving on tablet computers,” *Proceedings of International Conference of the Learning Sciences*, vol. 1, 2018.

[21] A. Solimeno, M. E. Mebane, M. Tomai, and D. Francescato, “The influence of students and teachers characteristics on the efficacy of face-to-face and computer supported collaborative learning,” *Computers & Education*, vol. 51, no. 1, pp. 109–128, 2008.

[22] S. Dewiyanti, S. Brand-Gruwel, W. Jochems, and N. J. Broers, “Students’ experiences with collaborative learning in asynchronous computer-supported collaborative learning environments,” *Computers in Human Behavior*, vol. 23, no. 1, pp. 496–514, 2007.

[23] H. Galperin, “ COVID-19 and the Distance Learning Gap,” http://arnicusc.org/wp-content/uploads/2020/04/Policy-Brief-5-final.pdf, Last accessed on 2020-07-28.

[24] S. J. Aguilar, “Guidelines and tools for promoting digital equity,” *Information and Learning Sciences*, 2020.

[25] L. Lipponen, K. Hakkarainen, and S. Paavola, “Practices and orientations of CSCL,” in *What we know about CSCL*. Springer, 2004, pp. 31–50.

[26] A. Weinberger, I. Kollar, Y. Dimitriadis, K. Mäkitalo-Siegl, and F. Fischer, “Computer-supported collaboration scripts,” in *Technology-enhanced learning*. Springer, 2009, pp. 155–173.

[27] T. Schellens, H. Van Keer, B. De Wever, and M. Valcke, “Scripting by assigning roles: Does it improve knowledge construction in asynchronous discussion groups?” *International Journal of Computer-Supported Collaborative Learning*, vol. 2, no. 2-3, pp. 225–246, 2007.

[28] “POGIL: Teaching online during the COVID-19 crisis,” https://pogil.org/teaching-online-during-the-covid-19-crisis, Last accessed on 2021-01-31.

[29] I. Howley, “Adapting guided inquiry learning worksheets for emergency remote learning,” *Information and Learning Sciences*, 2020.

[30] C. N. Gunawardena, C. A. Lowe, and T. Anderson, “Analysis of a global online debate and the development of an interaction analysis model for examining social construction of knowledge in computer conferencing,” *Journal of educational computing research*, vol. 17, no. 4, pp. 397–431, 1997.

[31] L. Gilbert and D. R. Moore, “Building interactivity into web courses: Tools for social and instructional interactions,” *Educational Technology*, vol. 38, no. 3, pp. 29–35, 1998.

[32] G. Gonzales, E. Loret de Mola, K. A. Gavulic, T. McKay, and C. Purcell, “Mental health needs among lesbian, gay, bisexual, and transgender college students during the covid-19 pandemic,” *Journal of Adolescent Health*, vol. 67, no. 5, pp. 645–648, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1054139X20304882

[33] M. West, G. L. Herman, and C. Zilles, “Prairielearn: Mastery-based online problem solving with adaptive scoring and recommendations driven by machine learning,” in *2015 ASEE Annual Conference & Exposition*. Seattle, Washington: ASEE Conferences, June 2015.

[34] T. A. Snijders and R. J. Bosker, *Multilevel analysis: An introduction to basic and advanced multilevel modeling*. Sage, 2011.

[35] M. S. Peteranetz and L.-K. Soh, “A multi-level analysis of the relationship between instructional practices and retention in computer science,” in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, ser. SIGCSE ’20. New York, NY, USA: Association for Computing Machinery, 2020, p. 37–43. [Online]. Available: https://doi.org/10.1145/3328778.3366812

[36] S. Poulsen, C. J. Anderson, and M. West, “The relationship between course scheduling and student performance,” in *Proceedings of the 4th Workshop on Educational Data Mining in Computer Science Education*, 2020.

[37] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2020. [Online]. Available: https://www.R-project.org/

[38] D. Bates, M. Mächler, B. Bolker, and S. Walker, “Fitting linear mixed-effects models using lme4,” *Journal of Statistical Software*, vol. 67, no. 1, pp. 1–48, 2015.