Evaluating an Adaptive Multi-User Educational Tool for Low-Resource Environments

Emma Brunskill
U. of California, Berkeley
Berkeley, CA
emma@cs.berkeley.edu

Sunil Garg, Clint Tseng
U. of Washington
Seattle, WA
skgarg,cxlt@cs.washington.edu

Joyojeet Pal
U. of Colorado & NYU-Poly
Boulder, CO & NYC, NY
joyojeet@gmail.com

Leah Findlater
U. of Washington
Seattle, WA
leakf@uw.edu

Abstract—**Quality primary education is a key focus in development. Educational software has the potential to help teachers provide good education. Due to limited computer resources in low-income schools, there has been substantial ICTD interest on multiple-user educational tools. Building on the foundation of research that addressed the technical problems and the human-computer-interface (HCI) concerns of multiple-input interfaces, our work on the educational game MultiLearn+ focuses on the potential educational benefits of multi-user systems. In contrast to prior ICTD multi-input educational designs, MultiLearn+ adapts the educational experience for each student within the group setting. This personalization may better meet individual students' needs. Our experimental results show that the adaptive capabilities of MultiLearn+ help even the competition between differing student backgrounds and abilities in a competitive game, increasing the probability that students will remain engaged and challenged.**

I. Introduction

Many development efforts focus on education due to the inverse correlation between education and poverty. Indeed, achieving universal primary education by 2015 is one of the United Nations Millennium Development Goals (MDGs). The scope of this MDG reaches beyond mere attendance, encompassing "quality education." [1] Ignoring for a moment the debate over how to measure or quantify education quality, it is clear that providing good education is a significant challenge throughout the globe. This challenge is typically amplified in developing regions where individual teachers are often responsible for many students and have few resources at their disposal.

One potentially useful resource a number of teachers do have is a computer; indeed, in India in 2009, over 14% of schools had at least one computer [2]. Though computers can be used primarily as resource-accessibility tools, such as compact libraries, another powerful use of computer technology is to provide automated tutoring and educational games. This educational software is traditionally designed for one-computer-per-child scenarios. However, in the developing region schools that do have computers, there are typically significantly fewer computers than children in a single class. This observation has given rise to substantial interest in the Information and Communication Technology for Development (ICTD) community on multi-input educational software (see e.g. [3]–[7]). Multi-user computer educational platforms have been used in multiple countries including Vietnam [8], the Philippines [9] and Belarus [10]. Despite this prior work, two important issues remain inadequately addressed. First, the research in this space has yet to articulate the extent of the learning benefit to students through significant empirical studies. Second, researchers have only begun to explore how the unique characteristics of the multi-user context can and should be utilized to support learning. In particular, the question of how to personalize each student's experience within the group setting has not been sufficiently explored. Our research goal was to expand the discussion on both issues, particularly on the topic of personalization.

Personalization has the potential to offer a number of benefits to students. Prior educational work has shown that one-teacher-per-student tutoring can be correlated with huge performance gains: one prior study reported that students who received one-on-one tutoring outperformed students who received standard classroom-only instruction by two standard deviations [11]. Though tutoring which pairs one teacher with one student is not financially nor logistically practical in most schools, educational software that personalizes each student's experience can provide some of the same benefits of one-on-one tutoring. For example, adaptive software can select different pedagogical activities for each student. Related to this, an exciting finding is that certain adaptive tutors have demonstrated especially large performance improvements for students with relatively weak initial skills (see [12], [13]). This suggests that adaptive tutors may be particularly helpful in catching up struggling students. Educational software can also monitor and track student progress and provide this feedback to teachers [14], [15]. This student assessment information can then allow teachers to better target their classroom-wide instruction, as well as further improve the effectiveness of their limited time for one-on-one interactions with individual students. These general benefits of personalization are potentially of even greater magnitude in low resource settings where teachers often have large classrooms with students of very widely varying backgrounds and abilities. For instance, with the increase worldwide in school enrollments across levels, some pupils may be of much older ages, due to having only recently received the opportunity for a free education.

One known challenge in ICTD multi-user setups is preventing a single student from dominating the computer session [3], [4], [7], [16]. Even when each child is equipped with his or her own input device, the problems of creating a more equitable learning environment persist. These problems arise because of unequal learning backgrounds and abilities, as well as the desire of some children to move at their own learning paces in simultaneous shared content scenarios.[1] Personalized multi-user educational tools may be built to allow each student to learn in different ways or at different speeds, and even to keep competitive scenarios on comparatively equitable grounds. This may lead to higher amounts of engagement amongst students, with the goal of facilitating long term learning gain. Indeed, there is some prior evidence (see for example Slavin, Leavey and Madden [17]) that students who receive personalized instruction enjoy and feel better about their performance in a given subject.

In this paper we present an adaptive multi-user educational game, MultiLearn+, designed for use in resource-constrained educational settings. Our initial trials in two low-income schools in Bangalore, India suggest that adaptive multi-user software can successfully account for students with differing skills in competitive educational games, which has the potential to keep each student more engaged and challenged. The software also monitors student performance, which can provide a valuable additional source of information to teachers.

In the rest of this document we discuss prior related work (Section 2), describe the software tutor used, and briefly outline the algorithms used to select activities adaptively for each student (Section 3). We then present the experimental setup (Section 4), our results (Section 5) and discussion (Section 6) before concluding (Section 7).

II. Related Work

A. Intelligent Tutors and Computer Aided Learning

There is an extensive literature on creating adaptive educational software for use by a single student (see [18]). Successful cognitive tutors such as the Adaptive Control of Thought (ACT) programming tutor described by Corbett and Anderson [19] have been found to improve student performance over standard classroom interactions by one standard deviation [20].

To our knowledge, there is very little published on using intelligent tutoring software or computer-aided learning software in developing world contexts. Mills-Tettey and colleagues [21] conducted a control trial of elementary students in Ghana who used an intelligent tutoring system designed to improve literacy. The authors found that students who attended a school in a low-income area and used the tutor as a supplement to classroom instruction outperformed students from the same school who only received classroom instruction. This was evaluated by fluency and the total number of words spelled correctly in pre and post test evaluations. However, there was no significant difference in post test results between students who did or did not use the tutoring system for students who attended a school in a wealthier area. The students in the higher-income area had significantly higher pretest scores than the other students, so it is also possible that intelligent tutors are particularly helpful for students who may be further behind. Indeed, this explanation is consistent with the findings of Beal and colleagues [12] and Sarkis [13], both which found that adaptive tutors were associated with larger gains in student performance for students with lower pretest performance. If lower-resource schools are generally associated with lower levels of student performance, as was present in the study by Mills-Tettey et al., this suggests that adaptive tutors may be of particular use in poorer educational settings.

Another study investigating the use of automed tutors in the developing world was conducted by Banerjee and colleagues [22] from the MIT Poverty Action Lab. The authors ran a large-scale experiment in conjunction with the Indian educational NGO Pratham. In this study, elementary school students were allocated either to a control group, or spent two hours a week in pairs at a single computer using computer-assisted learning (CAL) mathematics software. The authors conducted a two year study, with 55 schools serving as an experiment group, and 56 schools as a control. The authors found that students who used the CAL software outperformed students in the control condition by 0.35 standard deviations over the first year. This effect was larger (though the difference was not statistically significant) than an alternate intervention, called Balsakhi, in which local community tutors met regularly with groups of 15 to 20 students who were at the bottom of their class. The Balsakhi program was cheaper ($2.25/student/year) compared to the CAL intervention ($7.72/student/year), though this difference is likely to depend on the particular labor costs and local infrastructure conditions. Banerjee et al.'s work is generally encouraging as to the potential impact of computer-assisted learning in developing regions.

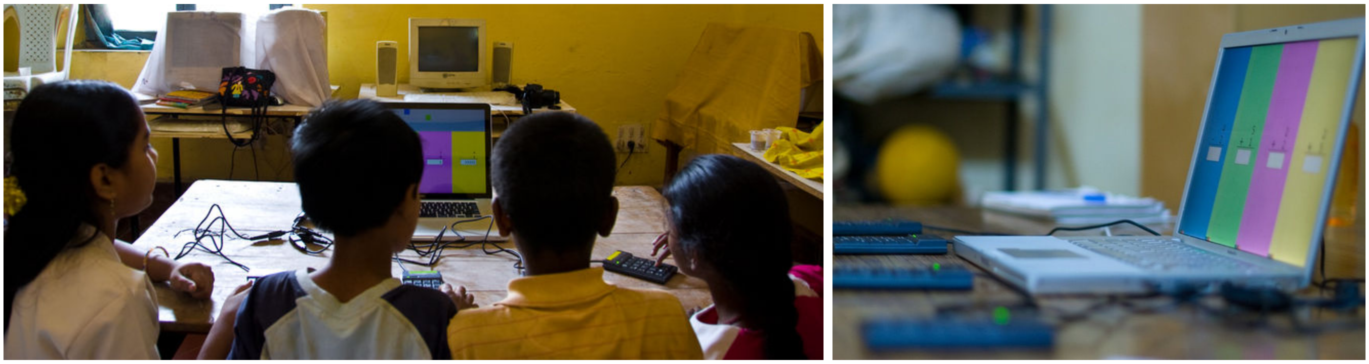
We have recently learned of parallel, ongoing work by Nussbaum et al. [23] that investigates the use of intelligent tutoring systems for many students in developing regions. One exciting aspect of this work is that it allows the teacher to interact easily with the students, as all students work individually on a single divided classroom screen. Our work has a different focus than their's, as our approach involves a competitive game amongst the students, and because we conducted a controlled trial where we considered no intervention.

B. Multi-user interfaces for developing regions

Within multi-user ICTD research, there has been considerable interest in the constraints that make some of this work distinct from the broader single-display groupware (SDG) literature. These include cost, ease of acquisition, and ease of maintenance of any additional devices.

Pawar and colleagues [3] investigated providing each student at a shared single computer with a mouse in a set of schools in Bangalore, India. The authors found that the students were more engaged when they each had a mouse, compared to a single-mouse condition in which students

[1] In Pawar, Pal and Toyama [3], the authors described a case where a student disliked one-mice-per-child interactions because it increased the competition amongst students, making it harder for individual students to contribute or work at different levels and speeds.



(a) (b)

Fig. 1. The MultiLearn+ system.

without a mouse typically became disengaged. Extending this work, the authors later conducted an English vocabulary retention task [4] and found that student performance gain from pretest to post test was not significantly different between students who were at their own computer versus students who each had their own mouse at a shared computer. This highlights the potential of multi-user software to provide an equivalent experience as single-user software. However, the pretest was conducted immediately before the session, and the post test immediately after, so the study did not measure longer-term interaction and retention effects.

Prior work has noted that even equipping each student with his or her own mouse does not eliminate the potential for a single student to dominate the interaction. Moed et al. [7] showed that imposing a procedural restriction on the interaction through the use of enforced turn-taking could reduce single-student dominance. Tseng et al. [16] considered explicit and implicit encouragement of collaboration by changing the incentives in a competitive game, and varying physical device sharing. In contrast to these two papers, which mostly rely on modifying the game rules to avoid dominance, in our work we propose to adapt the activities selected for each individual student's current progress to implicitly reduce the potential for dominance.

Beyond one-mouse-per-child with individual cursors on a shared screen, there have also been other interface choices explored for multi-user settings within the ICTD community. While mouse input is useful for many tasks, it is generally less well-suited to entering numbers or text (though see [5]). An alternative option that has been less explored in the multiple-input literature is to use keyboards or numeric keypads. In addition, rather than having all students work together on a single task, the screen can be divided into separate sections with different tasks (see e.g. [16]).

III. MultiLearn+

In this paper we provide a preliminary investigation of the benefit of creating multi-user software that can adaptively customize each student's learning experience in the context of developing world schools, thereby joining work on computer-assisted learning with the ICTD work on multi-user interfaces. As our motivation is to construct software targeted at low-resource communities, the underlying impetus for our work is not to facilitate collaboration, but rather how to best engage and challenge each individual student given that a one-computer-per-child situation is not feasible in the majority of our areas of interest. There is a rich set of questions (and some interesting work) regarding collaborative software for the sake of collaboration, and whether competitive learning, isolated learning, or collaborative learning are most effective, but in this particular study we leave these questions aside and focus on individual learning for students operating in a multi-user environment.

A. Core tutor

Our multi-user educational software builds upon the prior work on MultiLearn [16]. MultiLearn splits the screen into multiple regions, so that each student interacts with a separate part of the screen. This makes it well suited to interventions like ours where our interest is in customizing the activities provided to each student.

We extended existing MultiLearn software designed for simple mathematics drill exercises. The screen was split into 4 vertical regions, and math exercises were provided on each region. Each student received his or her own numeric keypad, which is associated with a particular region of the screen, and therefore, to a particular drill exercise. As an additional incentive to keep students engaged, a competitive aspect was introduced: students on the same computer compete to see who can correctly finish 12 questions the fastest.

The mathematics curriculum consisted of 19 skills. The general skill categories (addition, subtraction, multiplication, division and fractions) were selected based on math textbooks produced by the (Indian) National Council of Educational Research and Training. To construct the specific skills, we drew on work by Woolf [18] and Brown and Burton [24] that discuss common student errors. These authors also provide examples of the hierarchical structure between skills, namely which skills are typically needed before mastery of other skills are possible. For example, basic subtraction skills are needed in order to correctly do long division.

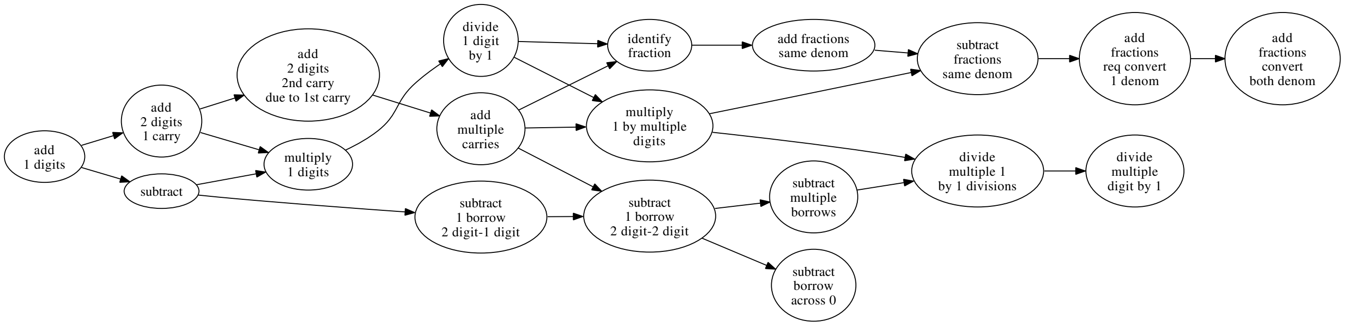


Fig. 2. Precondition graph showing hierarchical structure of skills supported by the MultiLearn+ tutoring system. Arrows indicate that the skill at the start of the arrow is needed for the skill at the end of the arrow.

The possible set of learning activities consisted of drill exercises of each of the 19 skill types. If a student answered incorrectly, the correct answer was displayed. In the future we hope to include a wider diversity of learning activities, such as hints and short lessons.

B. Individual adaptation

Our goal was to customize the experience for each student in order to keep each child engaged and challenged. One potential danger in multi-user educational software is that one student may dominate a session. In the past, dominance has been mostly discussed in the context of a single student maintaining control over a single input device, such as a mouse [3]. However, it is also possible that a single student will consistently win an educational game, or that certain students will find generic material much too challenging. Without individual customization, it is quite likely that this kind of scenario will occur, causing students to become bored, or struggle and give up, which can lead to disengagement and a poor learning experience.

To address this issue, we framed the problem of selecting which questions to provide a student within the artificial intelligence field of *decision making under uncertainty*. Briefly, the core intuition is that whenever a student answers a question correctly or incorrectly, that provides a small amount of information about whether or not the student understands the topic a question is probing. For example, if a student correctly answers that five times seven is 35, that provides some evidence that he/she understands one-digit by one-digit multiplication. This information is not a perfect indicator of whether a student correctly understands the question topic: even college math professors will occasionally make mistakes on basic addition but we do not question their fundamental understanding of addition. Conversely, a student may occasionally guess the correct answer without fully understanding the topic of the question, particularly if the student is selecting among multiple-choice answers.

Posing the problem of selecting each subsequent question for each student as an instance of decision making under uncertainty has multiple advantages. First, it provides a principled way for constructing a coarse estimate of the student's knowledge over the skills covered by the tutor. As the student receives questions about different topics, and answers each of them correctly or incorrectly, we can build up an estimate of the student's understanding of these topics. A simple way to do this would be to maintain a running count, or average, over all the topics the student has received questions about, and whether the student got that question correct or not. In fact, we can do better than this basic approximation by using two additional pieces of knowledge. First, as we mentioned previously, we anticipate that the student's performance is not a precise reflection of their true knowledge. By using information about how likely it is that a student answers a question incorrectly even if he/she knows the material, along with how likely it is the student will be lucky and guess the correct answer even if he/she does not understand the question topic, we can refine our estimates of which topics the student knows. Second, some topics build upon others. If a student answers multiple algebra questions correctly, it is highly likely that she also understands basic addition, subtraction, multiplication, division and fractions. By using knowledge of this structure among skills, which is known as a learning hierarchy in the education community (see e.g. [25], [26]), we can further improve our estimates of a student's understanding over a particular curriculum of skills. The topics covered in our software, and the hierarchical relationship between these topics, is depicted in Figure 2.

The second key benefit of this framework is that it can be used to formalize the process of selecting which exercise or tutorial to present to a student. Assuming that we start with little knowledge about which topics or skills a student knows, we want to balance between asking students to do exercises that we think will most help them progress towards mastering the full curriculum as quickly as possible, with questions that will help us diagnose which skills a student has understood and which skills the student needs assistance on. In a sense this division can be thought of as choosing between which exercises are most likely to help the student, and which exercises are most likely to help the teacher get a better understanding of what the student knows. This can in turn allow the teacher (and the automated tutor software) to better help the student learn.

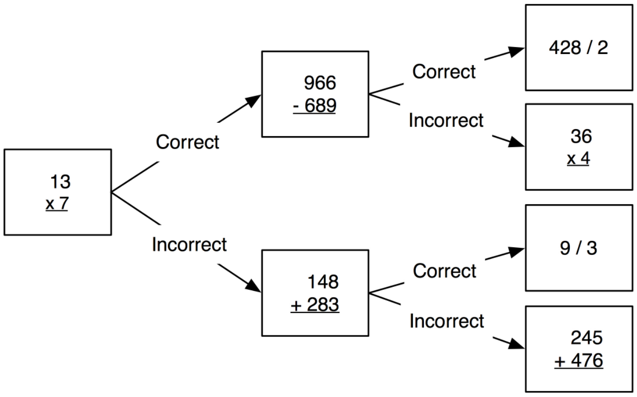


Fig. 3. The exercises selected depend on the prior exercises given and on how the student answered each of those prior exercises. Here we show a subset of the decision tree which displays which exercise to give next, conditioned on the prior student responses.

In our work we used the freely available partially observable Markov decision process (POMDP) planner Perseus [27] to both maintain an estimate of which skills a student understands, and to select the next question to provide to the student. The use of this software means that after all students receive a question on the same initial topic, each student will begin to receive different questions on different topics, based on which questions he or she answers correctly.[2] Figure 3 displays a flowchart which illustrates different exercises a student would receive, based on the prior exercises completed and past answers.

IV. Controlled Study

We now describe a controlled trial we conducted with two schools in Bangalore, India. The goal was to systematically investigate the potential educational benefit on student performance of using a multi-user AI-adaptive tutor, MultiLearn+. We operationalized this goal into the following three questions:

- Do students who use MultiLearn+ for multiple sessions improve their post test scores over their pretest scores?
- Is MultiLearn+ more effective than a non-adaptive software tutor (MultiLearn) at improving student performance?
- Are students who use MultiLearn+ more engaged than students who use non-adaptive software (MultiLearn)?

We are motivated here by the fact that there is relatively scarce evaluation of the benefit on learning outcomes of multi-user software, including MultiLearn. Given this, one of our additional interests was to evaluate whether the multi-user software was associated with learning benefits for the students, even in the non-adaptive condition. This required an additional control condition in the experimental design.

- One digit addition
- Two digit addition, multiple carries
- Four digit addition, multiple carries
- Two digit subtraction
- Two digit minus one digit, one borrow
- Two digit minus two digit, one borrow
- Three digit minus three digit, borrow across zero
- One digit by one digit multiplication
- Two digit by one digit multiplication
- One digit by one digit division
- Three digit by one digit division
- Express the shaded portion as a fraction
- Add two fractions, same denominator
- Add two fractions, convert one denominator
- Add two fractions, convert both denominators
- Add two fractions, convert both denominators, sum > 1

Fig. 4. Pretest and posttest questions.

A. Participants

The experiment was conducted in two schools in Bangalore, India that had previously been involved in MultiLearn prototyping studies. The first school (S1) was a public government school and the second school (S2) was a private school that was partially funded by the government, and which followed the goverment-provided curriculum. Both served low-income communities. Students in grades four and five participated in the study. We anticipated that students in these grades would be familiar with doing basic mathematics exercises and would therefore find the basic object of the tutoring software intuitive.

Due to the fairly small study size, and as it was anticipated that there were potentially significant differences between students in different grades, and between students in different schools, each school-grade was divided into each of the three conditions: control, MultiLearn or MultiLearn+. This is in contrast to large-scale economic randomized trials where typically entire schools are assigned to a single condition (for example, see [22]).

B. Experiment Design

To design the control condition, we considered several factors. First, we wanted to control against the possibility that students who received tutoring time would improve their performance simply due to the novelty of the software rather than any specific educational benefit of the software design. In other words, we were concerned about the Hawthorne effect, which is known to be associated with student post-intervention score improvements in education studies [28]. In addition, computer use is rare in the schools in which we conducted our experiments, and we wanted to give all students an opportunity to interact with a computer, to combat against feelings of unfairness amongst the students in the different conditions. Therefore students in the control condition used MultiLearn to play a multi-user spelling game instead of the math game. This gave all students equal opportunity for computer time. The allocation of students to each condition is displayed in Table I. Due to the size of the classes, and the multi-user focus of the study, it was not possible to have equally balanced

[2] In fact, if there is prior knowledge about the topics a student has mastered or is struggling with, this can also be encoded at the beginning. This means that even at the start each student may receive different question topics.

numbers of students in each condition. In these cases, we randomly assigned the additional groups to the adaptive and non-adaptive conditions, since those were our main conditions of interest. Beyond this preference, students were assigned randomly to groups.

The experimental design involved a paper pretest to evaluate initial student performance over the math skills outlined in the prior section, four sessions (over four separate days) on the computer using the condition-specific software, and a paper post test. Each computer session consisted of 30 minutes where the students were placed in small groups of up to four, each at a single computer. All MultiLearn and MultiLearn+ groups had four students except for one MultiLearn+ group which had two students.

The paper test consisted of 16 questions. Figure 4 shows the topic of each test question. Two versions of the paper test were constructed. Students were randomly assigned to receive one version of the test as their pretest, before any interaction with the computer software, and the other version for their post test. This was done to control against any unanticipated differences in difficulty between the two paper tests.

The experiment itself was conducted in schools, rather than a controlled laboratory, in order to best mimic how the software would be used in a routine, non-experimental fashion. Sessions were held in a small computer lab at School S1 (though lab computers were nonfunctional during the time of our visit). School S2 had no computers, and we re-purposed an available room for the computer experiments. All experiments were conducted on laptops brought to each session by the researchers.

As a wide variance in skill between students was expected, the same curriculum was used for both grades.

C. Data Collection

The system recorded all interactions with the adaptive and non-adaptive game, including the specific exercises provided to each student, the answers entered by each student, how long students spent on each exercise, when a game was won, and which student won the game. In addition, during all but the first study sessions, two experimenters also recorded observations of each group of students playing the adaptive and non-adaptive game. Specifically, each observer marked recorded instances of potential disengagement and conflict (abandoning the device, looking away from the computer and group, walking away, one student consistently winning, and criticism or negative verbal interaction) as well as collaboration (talking among group, pointing at screen and using another student's keypad). Unfortunately, no observations were able to be recorded on the first sessions due to initial set up tasks.

D. Data Analysis

There was no significant difference in pre-test performance between the two test versions (t(118)=1.88, p=0.063) and results were pooled across the test versions for the result of the analysis.

TABLE I
STUDENTS WHO TOOK PRETEST AND POST TEST

Grade	School	Number of children		
		Control	MultiLearn	MultiLearn+
Grade 4	S1	6	11	8
	S2	10	14	12
	Total	16	25	20
Grade 5	S1	8	11	9
	S2	12	12	12
	Total	20	23	21

TABLE II
STUDENTS WHO WERE ABSENT FOR THE PRETEST OR POST TEST

Grade	School	Number of children		
		Control	MultiLearn	MultiLearn+
Grade 4	S1	7	1	1
	S2	3	2	0
	Total	10	3	1
Grade 5	S1	2	2	1
	S2	1	0	0
	Total	3	2	1

A number of students were absent from school during either the pretest or posttest, and Table II shows the absences number per school, grade and condition. School S1 had many more student absences than school S2. In our subsequent analysis, unless otherwise stated, we only include students who completed both a pretest and a post test. Though there appear to be slightly more absences of students in the control group, we suspect that a condition-specific effect on student absence is unlikely, as the students generally get little exposure to computers, and were excited to participate.

When administering the pretest and post test, it was emphasized to the students that these tests were used to help us evaluate the effectiveness of our software, and would not be used to evaluate the individual students nor be passed on to their teachers. Even so, later analysis of the pretest and posttests revealed several students who appeared to have cheated with each other during the pretest and/or post tests.[3] We removed these students' data (2 fifth graders from MultiLearn condition, 2 fifth graders from MultiLearn+ condition) from the analysis of the paper pretest to post test score changes.

V. RESULTS

Figure 5 displays student pretest scores across grades and schools, and Table III displays the student pretest scores across conditions and grades. There was a large amount of variation among students in both grades, suggesting that the choice of using the same curriculum for the two grades was reasonable. The highest score was 68.75%: it is interesting to note that this student got all of the fraction addition questions correct, which the vast majority of students got wrong, but made errors on some of the questions which would typically be considered easier. Each question topic was answered correctly by at least one student, indicating that most of the curriculum covered in the pretest was appropriate for the students involved in

[3] Several student pairs had identical answers on all the tests questions, including the specific wrong answers entered.

TABLE III
STUDENT SCORES BY GRADE AND CONDITION. ALL SCORES ARE PERCENTAGES

Condition	Control				MultiMath				MultiLearn+			
	N	Pre	Post	Diff (σ)	N	Pre	Post	Diff (σ)	N	Pre	Post	Diff (σ)
4th Grade	16	32	39	7 (10)	25	37	42	5 (10)	20	38	47.4	9.4 (10)
5th Grade	20	42	46.7	4.7 (20)	21	35	41.5	6.5 (10)	19	44	43	-1 (10)

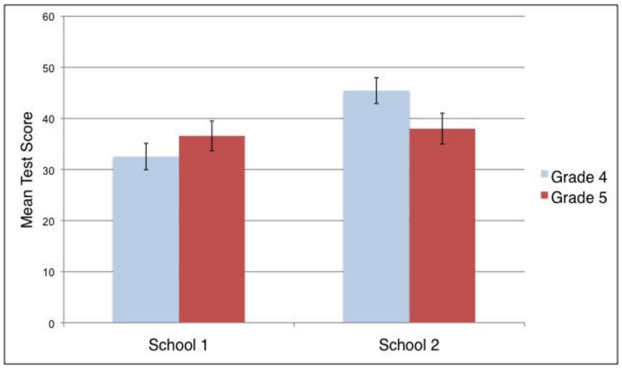


Fig. 5. Student Pretest Mean Scores. Error bars indicate 1 standard error. From left to right, the sample size was N=34, 33, 41, and 37, respectively.

the study. One potential exception to this were the fraction topics. Most fourth grade students gave answers that suggested they were completely unfamiliar with fractions (such as simply adding all the numbers in a fraction addition problem together), and many fifth graders were similarly challenged.

Though there was no significant difference between the pretest scores amongst students in different conditions, the mean pretest score for fifth grade students in the adaptive condition was higher than the scores of fourth grade students in the control condition.

We predicted the adaptive versus non-adaptive conditions would have different effects on student behaviors such as disengagement (e.g., walking away from the computer) and positive and negative interactions between students (e.g., one student criticizing another or two students collaborating on a question). Due to logistical reasons we could not record video logs of the school sessions, so instead our two observers recorded observations in situ. Unfortunately, no discernible differences arose from the data and, due to space constraints, we do not present the details here. We are, however, interested in doing a video analysis of student behavior in the future, which would allow us to have multiple coders and to assess inter-rater reliability.

A. *Results: Game Dominance*

We were interested in whether the adaptive software (MultiLearn+) helped reduce inner-group game dominance between students working on the same computer, compared to the non-adaptive MultiLearn tutor. To quantify game dominance within a group, we calculated what percentage of the time each student won a game in their group (recall that a game ends when the first student correctly answers 12 questions). Most groups completed several games over the four sessions, but two groups only completed a single game and were excluded from this analysis (which removed one adaptive group and one non-adaptive group). If we classify game dominance as when a single student wins 80% or more of the games, then the adaptive condition had half the instances of dominance compared to the non-adaptive condition (4 out of 14 groups compared to 8 out of 14 groups).

In some of these cases, so few games were completed that game dominance of a single student might be barely noticeable. For example, in one group, one student won once, and another student won four times, spread across the course of four sessions that were held on different days. Though this would fall under our previous definition of dominance, it is possible that in this situation students would not necessarily notice or be bothered by this infrequent but consistent winning of a single student.

To account for the frequency of a student winning the game, we also extracted instances where a single student won at least 10 more times than any other student in the group. Under this definition of frequent game dominance, we found there were 4 instances of game dominance (of 11, 12, 17 and 22 more wins) in the non-adaptive group. In contrast, there were no instances of this type of dominance in the adaptive condition.

Under both game dominance measures, the adaptive condition had many fewer instances of dominance compared to the non-adaptive condition. These results indicate that by selecting personalized questions for each individual student, the adaptive policy may provide a multi-user game that is equally challenging for each student. If so, this finding suggests that using an adaptive, individualized software may result in more equitable learning environment, which in turn has the potential to lead to higher levels of engagement for all students compared to the non-adaptive tutor.

B. *Results: Pre/Post Tests*

To quantify the difference between student performance on the pre and post paper tests, we used a repeated-measures unbalanced analysis of variance (ANOVA). An ANOVA provides a statistical test to evaluate whether there are significant differences between the means of multiple groups. In our analysis, School, Grade and Condition are the between-subjects factors, and Test (pre, post) is the repeated factor. There was a significant main effect of Test (F(1,97)=19.023, p $<$0.001), demonstrating that on average all students performed better on the post test compared to the pretest. There was no significant interaction between Condition and Test, indicating there was no systematic difference between the change in performance among students in the control, non-adaptive and adaptive conditions. However, there was a significant interaction between Condition, Grade and Test (F(2,97)=3.623,

$p < 0.05$), suggesting that there *was* a difference in test scores based on Condition but the effect was different for 4th grade than 5th grade. However, using the Tukey's honestly significant difference criterion to examine the pairwise mean comparisons (such as, 4th grade adaptive versus 4th grade non-adaptive) did not yield any significant results.

Nevertheless, this interaction reflects the trend, displayed in Table III, that fourth grade students showed the largest mean improvement in post test scores in the adaptive condition, whereas fifth graders in the adaptive condition showed no improvement. Fifth graders in the adaptive condition had the highest pretest scores, suggesting they had potentially less to gain from the tutor (Table III).

C. Results: Session Performance

We also examined the tutor session student data. Ultimately we are interested in whether the software is helping students to do a larger number of harder questions correctly. To quantify the relative difficulty of the questions that students answered during each session, we used the precondition graph structure from Figure 2 to assign each question of a particular skill type k a weight according to the number of precondition skills that skill k requires. For each student session, we computed the weighted sum of all the questions answered correctly by the student. We restricted our analysis to students who completed at least 3 sessions (N=95). We then ran an ANOVA with School, Grade and Condition (adaptive or non-adaptive) as between-subjects factors to examine if there were any group-wise differences in question difficulty between the first and last session.

There was no significant effect of Condition and no significant interaction terms with Condition. There are numerous potential explanations for this null result. One possibility is that the intervention was not significantly improving student performance. Another explanation is that the limited number of tutor sessions restricted the potential impact of the adaptive software compared to the non-adaptive software.

D. Results: Monitoring and Prediction

One of the potential strengths of posing tutor action selection as a partially observable decision making under uncertainty process is that it provides a principled approach for estimating and predicting student progress. We were interested in evaluating how well the final estimate of student knowledge computed by the adaptive tutor at the end of the computer sessions corresponded to the same student's post test performance. One challenge is that the estimate computed by the tutor represents a probability distribution that a student knows each of a set of topics, whereas the post test represents a single sample of whether the student happened to answer each topic correctly. This implies we would not generally expect the post test to be a perfect representation of the student's set of acquired skills. Using the estimate provide by the tutor, we predicted a post test score by rounding the computed probabilities for each topic on the paper test. For example, if a student was estimated to have mastered 2-digit division with 80% probability, we rounded this value to predict that this student would answer the 2-digit division question correctly on the paper test.

The Pearson correlation between the predicted post test scores and true post tests scores was r=0.56. This suggests that the adaptive tutor is capturing useful information about a student's progress.

VI. Discussion

Our objective in undertaking this research was to advance the still nascent exploration of interfaces that customize each student's experience within a group environment, and to evaluate the educational benefits of multi-point tools. Our results underscore the potential benefit of adapting multi-user interfaces to each individual user, demonstrating that an adaptive interface was associated with substantially fewer instances in which a single student consistently won the game. This in turn has the potential to keep students challenged and engaged over longer time periods, which is likely to lead to higher performance outcomes and increase positive attitudes about the subject material. In terms of our second objective, the results of this study do not yet support a case for the educational benefit of this particular multi-user software. We comment on this issue and several others further below.

There are several important limitations of the empirical study that should be considered when attempting to generalize its results more broadly. First, our software was designed to provide practice over a fairly large set of elementary mathematics skills, but we only conducted a four-session study. Though this is longer than multiple prior multi-user ICTD studies, our study is shorter than many broad-scope education controlled experiments, which frequently take place over multiple weeks to months (see e.g. [22]. In contrast, some intelligent tutoring systems (ITS) have found significant effects over short interventions, but some such systems test for a much more specific result or focus on more on whether students' problem-solving process changes as a result [29]. In addition, absenteeism was a significant issue, and occurred both during the software sessions and on the test days. In fact, only 68% of students attended all four sessions. This suggests that the lack of a significant difference in student performance amongst the experimental conditions should not be considered definitive, and that further study is required.

Our experience also suggests that a closer collaboration with local teachers would improve the tutor curriculum development. Despite fractions being covered in an Indian textbook for our targeted student age group [30], the students' paper test performance demonstrated that almost all students had no understanding of fractions. To ensure no bias was introduced due to unfamiliarity with the symbolic operators chosen, we confirmed that locally consistent symbols were used.[4]

[4] The initial format we used to describe division on the pretest turned out not to be the common symbol used: however, we discovered this early on, and corrected all subsequent papers to use the local symbol. This error affected only the pretests scores of the fourth graders in one school. A t-test comparing fourth grade pretest scores on the two division test questions across the two schools showed no significant difference (t(105)=0.3652, p=0.7157) so we did not explore this issue further.

However, as our software did not provide any tutorials or hints, if a student was unfamiliar with a topic, or needed significant assistance, it is unlikely the student would master the material simply through trying to do drill exercises, failing, and seeing the right answer. Many intelligent tutoring systems provide a range of pedagogical material, including within-question assistance through the form of hints and other guidance. This type of scaffolding could be very helpful in supporting student progress. Another more advanced form of adaptation that we believe would be helpful is more in depth analysis of students' incorrect responses. Past work from the educational psychology community [24] has established that student errors are typically not random, and are instead often the result of systematic errors in understanding. The evidence from our own student study certainly supports this theory: for example, when faced with a fraction addition question they did not understand, students often either added up all the numbers (numerators and denominators) to generate a final number, or else added all the numerators and divided that by the sum of all the denominators. Though both procedures are incorrect, they are both completely rational approaches that exploit differing amounts of prior knowledge of the student. Analyzing such student errors in more depth could help better adapt the software to each student.

There were several instances where a single student helped other students repeatedly at the same computer. This suggests that the competitive aspect of the game is not distracting from group collaboration: it is unclear whether the competitive game aspect of the tool is necessary or helpful. However, this observation does suggest that it might be interesting to explicitly include within-group collaboration or assistance as a potential tutor action. Indeed, as the software monitors each student's progress, the software could instruct one student to ask for help from another specific student who has already mastered that topic. This also opens up the possibility of creating tools that customize and adapt the tutoring experience to the full student group, rather than just to each individual student. For example, if the system identified that all students at the computer are struggling with two-digit multiplication, the system could switch into a single-screen mode and provide a short video or tutorial to all students about two-digit multiplication. Group adaptation also offers the possibility of directly trying to encourage collaboration, which may further motivate and encourage students. It also may be helpful to directly model student motivation, and select tasks to maintain a high level of motivation.

When observing students using the software, several teachers commented that they were surprised by particular students' performances. This supports the intuition that personalized software that monitors student progress can serve as an additional form of feedback to the teachers about the students, without putting students into explicit and potentially stressful "test-taking" mode, where they may resort to cheating, as appeared to occur with our own exams.

Finally, a brief discussion of costs and scalability is important given the limited resources present in many developing regions. Computers are already present in a number of schools, and this trend is likely to continue. Though perhaps slightly less commonly available than computer mice, numeric keypads are nearly universally compatible, low cost (approximately four US dollars) and offer easier numeric input than mice, with a much smaller form factor than keyboards. This last property is a particular benefit since multiple students are sharing the same computer and screen. Schools do not typically have enough computer mice for each student, so equipping each student with an input device will generally require an additional investment in computer accessories, whether it be a numeric keypad or computer mouse. Beyond this upfront investment, software can be provided at low cost or even free, which improves the scalability potential of multi-user software, and suggests it may be a helpful tool to low-resource school systems seeking to further supplement their tools to bolster student performance.

VII. Conclusion

Rooted in a well-established premise of inadequate access to computing because of machine-sharing, multiple-input formations have provided an important and consistent area of work within ICTD research. Our goal with the research presented in this paper was to expand the discussion on the longer-term educational benefit of multi-user educational tools, and the range of possibilities offered by individualizing the software to each user. We have shown both the usability of an adaptive system as well as its benefits in keeping children competing on comparatively equitable grounds. The approach of splitting screen resources and adaptively pushing different questions to individual input devices also has the potential to lead to a better estimate of individual student progress, which may provide valuable input to teachers. In summary, we believe adaptive, multi-user software has promise as a cost-effective and scalable tool towards improving education in developing regions.

VIII. Acknowledgements

The authors gratefully acknowledge the assistance of Meera Lakshmanan in assisting with local arrangements for this study, and thank Richard Anderson for his support. E.Brunskill was supported by a NSF Mathematical Sciences Postdoctoral Fellowship. C.Tseng and S.Garg were supported in part by the National Science Foundation. J.Pal was supported by the National Science Foundation (NSF) under Grant #0937060 to the Computing Research Association for the CI Fellows Project. L.Findlater was supported by a NSERC Postdoctoral Fellowship.

References

[1] United Nations, "The millennium development goals report," Tech. Rep., 2008.

[2] N. U. of Educational Planning and Administration, "Elementary education in india: Progress towards uee," http://www.dise.in/Downloads/Publications/Publications%202008-09/Flash%20Statistics%202008-09.pdf, Tech. Rep., 2010.

[3] U. Pawar, J. Pal, and K. Toyama, “Multiple mice for computers in education in developing countries,” in *Proceedings of the International Conference on Information and Communication Technologies and Development (ICTD)*, 2006, pp. 64–71.

[4] U. Pawar, J. Pal, R. Gupta, and K. Toyama, “Multiple mice for retension tasks in disadvantaged schools,” in *ACM Conference on Human Factors in Computing Systems CHI*, 2007, pp. 1581–1590.

[5] S. Amershi, M. Morris, N. Moraveji, R. Balakrishnan, and K. Toyama, “Multiple mouse text entry for single-display groupware,” in *Proceeding of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, 2010.

[6] K. Heimerl, D. Ramachandran, J. Pal, E. Brewer, and T. Parikh, “Metamouse: Multiple mice for legacy applications,” in *ACM Conference on Computer-Human Interaction (CHI), Work in Progress*, 2009.

[7] A. Moed, O. Otto, J. Pal, U. Singh, M. Kam, and K. Toyama, “Reducing dominance in multiple-mouse learning activities,” in *Proceedings of Conference on Computer Support for Collaborative Learning (CSCL09)*, 2009.

[8] Microsoft, “Microsoft multipoint pilot case study,” http://download.microsoft.com/download/8/0/F/80F567C1-214D-404C-A4CA-C09366B2E409/MouseMischief_CaseStudy_LeQuyDon.pdf, Tech. Rep.

[9] ——, “Multipoint helps ignite passionfor learning in the Philippines,” http://download.microsoft.com/download/A/3/1/A31394D6-DED3-4D35-8A3A-0BCCBEBC7189/MultiPoint_Phillipines_Case_Study_April09.pdf, Tech. Rep.

[10] ——, “Teachers optimize interactivity in the science classroom with windows multipoint mouse sdk from microsoft and ael collaborative content,” http://download.microsoft.com/download/F/4/3/F436C1BB-6186-4728-890C-A501E8899C47/Windows_MultiPoint_SDK_Pilot_Case_Slveco_Belarus.pdf, Tech. Rep.

[11] B. Bloom, “The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring,” *Educational Researcher*, vol. 13, no. 6, pp. 4–16, 1984.

[12] C. R. Beal, R. Walles, I. Arroyo, and B. P. Woolf, “Online tutoring for math achievement: A controlled evaluation,” *Journal of Interactive Online Learning*, vol. 6, pp. 43–55, 2007.

[13] H. Sarkis, “Cognitive tutor algebra 1 program evaluation: Miami-dade county public schools,” Lighthouse Point, FL: The Reliability Group, http://www.carnegielearning.com/web_docs/sarkis_2004.pdf, Tech. Rep., 2004.

[14] E. Kosba, V. Dimitrova, and R. Boyle, “Adaptive feedback generation to support teachers in web-based distance education,” *User Modeling and User-Adapted Interaction*, vol. 17, pp. 379–413, 2007.

[15] E. Gaudioso, F. Hernandex-del Olmo, and M. Montero, “Enhancing e-learning through teacher support: Two experiences,” *IEEE Transactions on Education*, vol. 52, no. 1, pp. 109–115, 2009.

[16] C. Tseng, S. Garg, H. Underwood, L. Findlater, R. Anderson, and J. Pal, “Examining emergent dominance patterns in multiple input based educational systems,” in *Interaction Design for International Development*, 2010.

[17] R. Slavin, M. Leavey, and N. Madden, “Combining cooperative learning and individualized instruction: Effects on student mathematics achievement, attitudes, and behaviors,” *The Elementary School Journal*, vol. 84, no. 4, pp. 408–422, 1984.

[18] B. Woolf, *Building Intelligent Interactive Tutors, Student-Centered Strategies for Revolutionizing E-Learning*, 2008.

[19] A. Corbett and J. Anderson, “Knowledge tracing: Modeling the acquisition of procedural knowledge,” *User Modeling and User-Adapted Interaction*, vol. 4, pp. 253–278, 1995.

[20] K. Koedinger, J. Anderson, W. Hadley, and M. Mark, “Intelligent tutoring goes to school in the big city,” *International Journal of Artificial Intelligence in Education*, vol. 8, pp. 30–43, 1997.

[21] G. Ayorkor Mills-Tettery, J. Mostow, M. Dias, T. Sweet, S. Belousov, M. Dias, and H. Gong, “Improving child literacy in africa: Experiments with an automated reading tutor,” in *Proceedings of the International Conference on Information and Communication Technologies and Development (ICTD)*, 2009, pp. 129–138.

[22] A. Banerjee, S. Cole, E. Duflo, and L. Linden, “Remedying education: Evidence from two randomized experiments in india,” *Quarterly Journal of Economics*, vol. 122, pp. 1235–1264, 2007.

[23] M. Nussbaum, C. Alcoholado, A. Tagle, F. Gomez, F. Denardin, H. Susaeta, M. Villalta, and K. Toyama, “One mouse per child: Interpersonal computer for personal formative assessment,” in submission.

[24] J. Brown and R. Burton, “Diagnostic models for procedural bugs in basic mathematical skills,” *Cognitive Science*, vol. 2, pp. 71–192, 1978.

[25] R. Gagné and L. Briggs, *Principles of Instructional Design*. Holt, Rinehard, and Winston, 1974.

[26] J. Close and F. Murtagh, “An analysis of the relationships among computation-related skills using a hierarchical-clustering technique,” *Journal for Research in Mathematics Education*, vol. 17, no. 2, pp. 112– 129, Mar. 1986.

[27] M. Spaan and N. Vlassis, “Perseus: Randomized point-based value iteration for POMDPs,” *Journal of Artificial Intelligence Research*, vol. 24, pp. 195–220, 2005.

[28] R. Clar and T. Sugrue, “Research on instructional media, 1978-1988,” in *Instructional technology: past, present, and future*, G. Anglin, Ed., 1991, pp. 327–343.

[29] C. Conati and K. Muldner, “Evaluating a decision-theoretic approach to tailored example selection,” in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

[30] *Math Magic. http://www.ncert.nic.in/textbooks/testing/Index.htm.* Indian National Council of Educational Research and Training.