Actively Engage Students with Diverse Background Using a More Personalized Approach

Jiaofei Zhong
Department of Computer Science
California State University East Bay
Hayward, California, United States
jiaofei.zhong@csueastbay.edu

***Abstract*—This Research to Practice Work in Progress paper presents the design of personalized learning modules on adaptive learning platform in order to engage students of diverse background in both online and face-to-face classes. Although the theory of adaptive learning approach has been widely discussed recently, there is no effective tool or course materials available to use directly in the bottleneck course of Data Structures and Algorithms, in which the DFW rate is much higher and the passing rate is much lower, especially when students have diverse background and various programming skills. Considering the dissimilar needs from students, we have redesigned this course using adaptive learning platform called Smart Sparrow, and created the weekly adaptive learning modules embedded into Blackboard LMS. The modules could be easily applied or modified for other computer science courses as well. Additional tools like Open Educational Resources and ZyBooks, as well as group-based assignments and projects are also utilized to further assist the students. Our goal is to evaluate the effectiveness of developed modules and student performance in this redesigned course. Preliminary results and assessment findings are discussed at the end of this paper.**

Keywords—adaptive learning, smart sparrow, engagement

I. INTRODUCTION

Recently, the adaptive learning approach has been studied intensively, which denotes self-paced learning with allocated resources based on the unique needs of individual learner [1]. The course of "Data Structures and Algorithms" (DSA) is an upper division course in Computer Science (CS) program, where the "DFW rate" (percentage of grades of D, F, and W) is usually higher than many other CS courses. This project adopts the following pedagogical strategies: adaptive learning [2], flipped classroom [3], supplemental instructions (SI) [4], project-based learning [5], and team-based activities such as pair programming [6]. By developing **adaptive lessons** on **Smart Sparrow** (SS) platform [7] that automatically adjust the learning pace for students with diverse background, we expect that students will attain fundamental programming skills and improve problem solving capabilities, which are essential for them to succeed in their future careers.

II. COURSE BACKGROUND

The DSA course is highly demanded by CS majors and minors, not only because it is a required core course in CS undergraduate program, but also a prerequisite of many other CS courses. It builds necessary foundation for CS students to pursue future careers in technology fields after graduation.

Supported by California State University Chancellor's Office CRT grant, and National Science Foundation under award number DUE-1626612.

A. Course Characteristics and Challenges

This course is a required upper division CS course. Students must pass this course with a grade of C- or better, in order to enroll in other courses that have it as prerequisite. This course contains extensive technical contents, and requires concrete programming skills. Hands-on projects for practice are necessary, which are also time-consuming and difficult for some students not good at programming. Our **main goal** is to develop adaptive lessons to help students who have difficulties learning this subject. The typical class capacity is 35 students per section, where on average 5 students receive grades below C- or withdraw during each term, thus the DFW rate is usually higher than 10%. In order to help students learn DSA topics, understand theories, and implement applications, we developed the adaptive lessons and SI sessions, flipped the classes, and organized team-based activities. We expect that students can achieve essential programming skills and improve problem-solving capabilities from these activities, which is essential for them to succeed in their future career.

B. Student Background and Characteristics

Students taking this course are mainly CS majors or minors, whose programming skills are not sufficient. In each term, there are both undergraduate and graduate students taking this course as required or for admission prerequisite, where majority of the undergraduate students are transferred from community colleges and take it as the first course here, meanwhile many of them are working part-time or full-time after school. Transferred students often lack the necessary programming skills, probably because they learned a different programming language at their previous institute, and the programming projects are difficult for them to finish. Both SI sessions and adaptive lessons are beneficial to this group of students with diverse background and skills.

TABLE I. UNDERGRADUATE STUDENT BODY PROFILE (FALL 2014)

Number of Students	
Headcount: 12,242	Full Time: 10,663 (87%)
Gender (per headcount)	
Men: 4,637 (38%)	Women: 7605 (62%)
Racial/Ethnic Groups (% of undergrads)	
Latino: 3,484 (28%)	Pacific Islander: 135 (1%)
Two or more races: 732 (6%)	Black/African American: 1,359 (11%)
Asian: 2,910 (24%)	Native American: 23 (<1%)
White: 2,309 (19%)	Unknown: 529 (4%)
Nonresident: 761 (6%)	
Low Income Students	6,495 (61%)
Mean Age of Students	24 years old
First Generation Students	5,930 (56.9%; 13.3% unknown)

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

Meanwhile, there are advanced students in each class, who can learn new concepts fast and complete the projects without any problem. Therefore, adaptive learning will be beneficial to this group of students, since they can skip some known topics, and spend more time on advanced topics and projects. Table 1 profiles our students, where 56.9% are first generation college students, and 61% are low-income students. In addition, our campus has 40.7% of its population designated Latino.

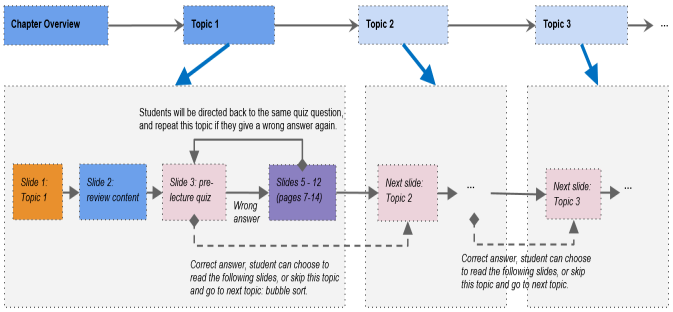

Fig. 1. Design Flow Chart of an Adaptive Lesson on Smart Sparrow

III. Course Redesign

This project aims at enhancing students understanding of the concepts of DSA, as well as providing more programming projects to promote students' problem-solving skills. Through the application of flipped classroom and SI sessions, as well as team-based activities and pair programming, this redesigned course will emphasize on the development of programming skills and analytical abilities necessary to design and develop computer-based solutions to complex problems. Students are expected to receive extensive instructions to gain programming experience such that they can succeed in this course and prepare for their future career.

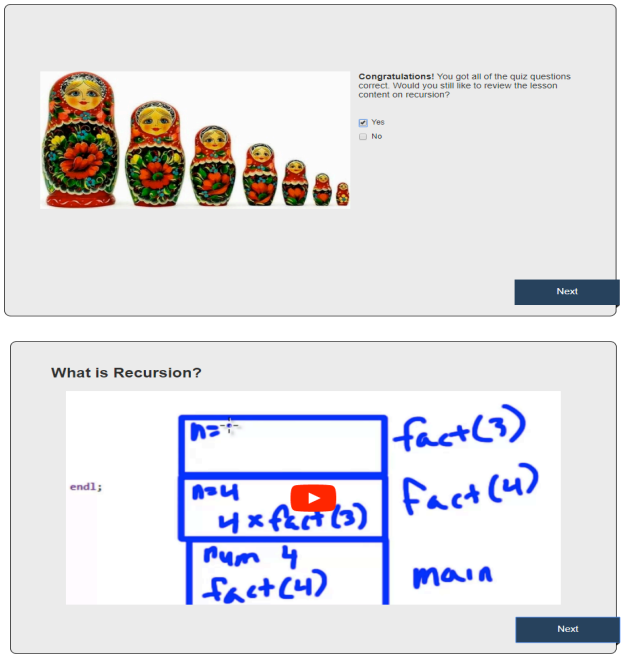

Fig. 2. Content Pages of Redesigned Adaptive Lesson on Smart Sparrow

A. Course Redesign Approaches

The traditional lecture consists of many definitions and concepts, followed by a few coding examples, since there are too many topics to cover within limited time. Now the course is redesigned with the following instructional pedagogies:

1) Adaptive Learning: SS platform is adopted to assist the instructors developing adaptive learning modules according to SLOs of this course. Each adaptive learning module contains a number of topics, which is designed for one of the SLOs. In each topic, there is a pre-lecture quiz consisting of a few multiple-choice (MC) questions, to be completed before each face-to-face lecture. According to the student's performance in pre-lecture quiz, they can skip the course content of this topic if they answer all MC questions correctly, and directly go to the next topic or still review the current topic's content pages if they like. Otherwise, if they gave any wrong answer, they have to review all pages of the course content of this topic, and then continue to the next topic, starting with a new pre-quiz for that new topic. After each lecture, students complete a post-lecture module with post-quiz containing the same set of MC questions as in the pre-quiz, which is used to test their SLOs for the same topic after each face-to-face class.

2) Flipped Classroom: The first half of class consists of mini-lectures on topics that students didn't perform well in pre-quiz, as well as important topics that students may be confused about in that Chapter. During the second half of class, in-class activities and group-based projects are implemented for students to practice.

3) Project-Based Learning: Every week there is at least one programming project designed for each lecture. Students must solve the programming problems in groups during the class. The first two groups that finished the in-class project with correct outputs can earn extra credit. In addition, there will be one more project as homework assignment per week.

4) Pair-Programming: The pair-programming strategy is introduced to the programming projects, when students are paired as a group to work together on one computer, and partners of the paired groups have specific roles of program development. For example, one member write the code, and the other member review it and point out errors. Students switch the roles with partners for each lab problem.

5) Supplemental Instruction (SI): Students are hired to lead the SI sessions, who have passed this course in a previous term with a grade of B or better. For one hour per week, they lead the discussion sessions and help students currently taking DSA class with difficulties. Almost every student taking DSA attended the SI sessions, especially those before the exams.

B. Adopted Technologies

We use **Blackboard** LMS (Learning Management System) for course management, which provides space to accommodate all teaching materials, including the syllabus, lecture notes, announcements, as well as assignments, tests, projects, and quizzes. **SS** platform has been adopted to develop the adaptive learning modules, which can be embedded onto Blackboard. Therefore, the students' grades will be extracted automatically from and entered into the Grade Book of Blackboard. The main

advantage of using SS is that it provides the platform to build adaptive learning content and share the materials with other instructors easily, and offers data analysis and generates charts to show students performance. We can also embed a series of **videos** related to the course content in those adaptive modules. It is often easier for students to learn via watching short videos with demonstrations and examples, compared against reading technical content or research papers. The redesigned adaptive lesson's flowchart is illustrated in Fig. 1, and two redesigned content pages of adaptive lessons on SS are shown in Fig. 2. In addition, **ZyBook** has been utilized as extra resources to help students learn, which contains plenty of interactive activities and animations to demonstrate the concepts and definitions, and provides statistics of student performance.

C. Assessments to Evaluate Student's Achievements

Students' achievements in terms of the Student Learning Outcomes (SLOs) are evaluated in the following assessments: **Formative Assessment** (30%) measures how well students perform during the learning process, including the in-class team-based projects, quizzes, written homework and programming assignments covering all SLOs, in order to help instructors adjust teaching and learning activities to improve students' class participation, engagement, and achievement. **Summative Assessment** (70%) measures how far students have achieved of the intended SLOs at the end of learning process, including mid-term exam and final exam, where questions are skill-based to evaluate student's capability in designing and implementing the DSA to solve problems.

D. Accessibility, Affordability, and Diversity Considerations

The pedagogical strategies support student's learning with diverse backgrounds, with the consideration of cultural, ethnic, gender, student learning style preferences, socioeconomic status, and first generation students, etc. Affordable learning solutions and "Open Educational Resources" (OER) [8] were included in this redesigned course. A search for OER resources has been conducted on various websites like COOL4Ed, MERLOT, and OpenDSA [9], etc. One free ebook [10] is found available and is selected as the required textbook for this course. Additional interactive ebooks such as ZyBooks [11] are recommended as well. During each term, the estimated cost of course materials with adoption of OER is approximately $50 per student, comparing against the typical cost of $218 per student. Thus, the **cost saving** is $168 per student per term, or 77% saving when using more affordable learning materials.

IV. Results

A. Impact on Teaching and Learning

This project had great impact on both instruction and learning, which involves a major restructuring of the bottleneck course of DSA. The adaptive learning modules are developed on SS platform, with built-in videos as well as pre- and post-lecture quizzes, interactive eBooks such as zyBooks, flipped classroom, and SI sessions. This project has strengthened the instruction by allowing instructors to explore various instructional technologies, and efficiently utilize class time on important topics that are more difficult for students to learn. In addition, this project has a positive effect on student's learning activities, since students have more opportunities to practice through the interactive learning modules, which in turn increased their motivation towards learning key concepts of the course. Thanks to the project-based learning and group-based activities, students are more engaged in class. In addition to the quantitative findings, we have observed that students ask more questions in the redesigned version of the course. This outcome is in-line with the goal of this project, which aims at increasing students' success via enhancing their engagement in class with the help of advanced instructional pedagogies that may increase their interests on the course content. Since it improves their motivations to learn and practice, this project allows more students to achieve better understanding of the DSA concepts.

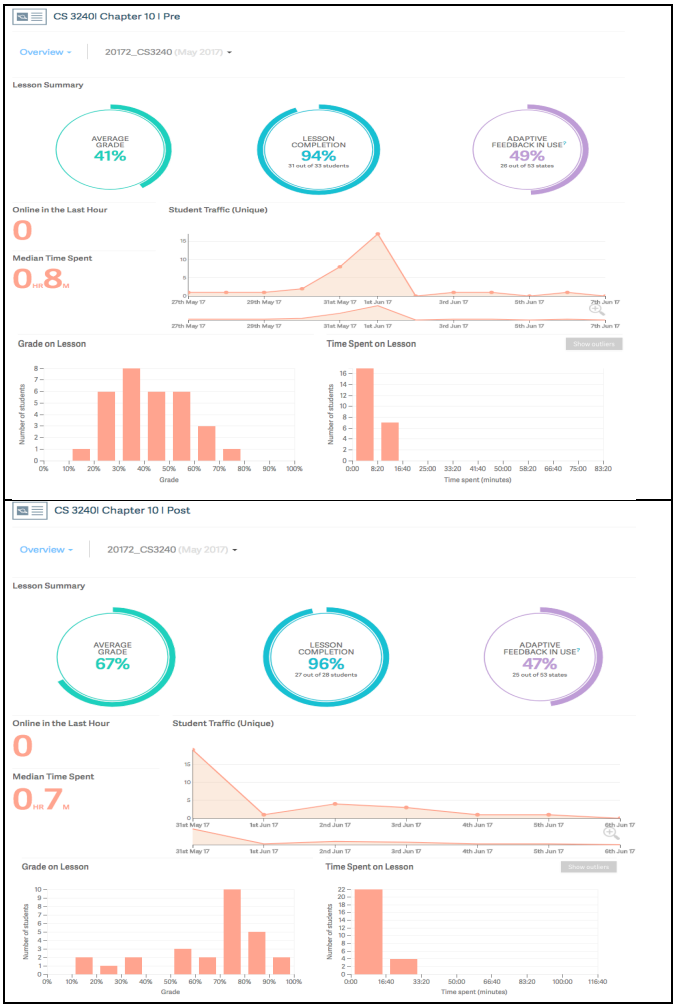

Fig. 3. Comparisons of Pre-Quiz and Post-Quiz Student Performance

B. Assessment Findings

Pre-lecture quizzes and post-lecture quizzes are created for each topic, in order to evaluate the performance of students. Each quiz contains 2-9 MC questions, built in SS modules or directly created on Blackboard. Pre-lecture quiz was conducted at the beginning of each week before lectures of each chapter, and post-lecture quiz was conducted at the end of the class of that chapter, to test the students' knowledge and skills after learning. The comparisons between pre-quizzes and post-quizzes can be used to analyze the teaching and learning

effectiveness on various topics, and an example is shown in Fig. 3, from which we can see that the **average grade has increased** from 41% in pre-quiz to 67% in post-quiz. Then, Fig. 4 shows the comparison of students' final letter grade in pre-redesign and post-redesign DSA courses: The first set compares a pre-redesign course in Fall 2014 of 29 students, against the redesigned course taught in Spring 2017 of 35 students; The second set compares a pre-redesign course in Spring 2016 of 27 students, and the same redesigned course in Spring 2017. We can see that in the redesigned class, the number of "A" students dramatically increased to 16, and the number of students receiving "D" or "F" dropped to 0. The number of students with **passing grade** ("C-" or better) has increased intensely and almost **doubled** in the redesigned class.

C. Student Feedback

A survey for this course redesign project was sent to students at the end of the quarter, in order to gather information and feedback about the students learning experience. According to the survey results, comparing this redesigned class with other classes, 42.9% of the students read the required textbooks and materials more, probably because of the embedded videos, adaptive learning modules, and interactive participation activities. Students commented that: "The awesome part about the online textbook is that it's interactive." "Instead of simply looking at the pictures within the text, I was able to interact with them, and practice the concepts that I was learning." "It made reading the textbook fun!" "Very helpful." "it was really easy to go to the chapter I wanted to read."

It is the first time that we adopt the SS platform to develop the adaptive learning lessons, which is entirely new to the instructors and every student in the class. There were some technical challenges for everyone to learn how to use the platform, and for the developer team to learn how to create new adaptive learning lessons, how to embed those lessons onto Blackboard, and how to automatically link the two platforms in order to send the grades from SS to Blackboard Grade Center. Fortunately, both instructors and students get familiar with the new technology quickly.

D. Lessons Learned

1) Teaching Tips: SS and the other platform's training workshops will be helpful to other instructors. The adopted technologies and adaptive learning modules developed for this redesigned course will be shared with other instructors who want to teach this redesigned course in the future. A sample syllabus and weekly study plan is developed and posted online before the first week of class, which can be shared with other instructors as well. When videos are embedded in the course materials, shorter videos within 8 to 10 minutes are recommended. Bonus problems are useful for fast learners and advanced students too.

2) Course Redesign Obstacles: Class meeting time for interactive activities is limited. This redesigned course will work better if the class is longer than 75 minutes. Some students didn't complete the pre-lecture quizzes before the class, even if there is a deadline posted online.

3) Teaching Strategies: In order to increase student engagement in class, we utilize flipped classroom and group-based, team-based projects to improve their engagement. There are also in-class discussions where the instructor drops questions to students for discussion, in-class demonstrations where the instructor performs hands-on coding demos to draw students attention, and in-class activities where the instructor designs an interactive activity for everyone to participate, for example a game in which students play together to simulate some algorithm's execution.

4) Instructor Reflection: During this academic year, I learned a lot by participating in all activities for the course redesign project. I redesigned this course step by step, applying the new technologies, creating the adaptive learning modules with the help of the "online campus", developing e-Portfolio in four stages, attending the training workshops, and participating in the "Professional Learning Community" webinars and discipline-based cohort meetings. All of these activities greatly helped me to improve teaching skills for all my courses. Some of the teaching challenges I encountered before have been solved. Basically, I incorporated a series of hands-on projects into the classroom to enhance students' engagement. One challenge is that the in-class projects took away the lecture time. Usually I had to skip some topics from the curriculum to accommodate theses activities. Now, flipping classroom can solve this problem. Second, I learned lots of advanced instructional pedagogies and new techniques, which can be applied to other courses easily.

5) Advice for Students to be Successful: Students should complete the pre-lecture modules with quizzes before class for a better learning experience. During the lectures, students should participate in all activities and team projects, and try to apply the knowledge into real-world applications. After class, students should complete both post-lecture modules and assignments. The total amount of hours each students need to spend on this course may vary, but the suggested amount is at least 10 hours outside the classroom.

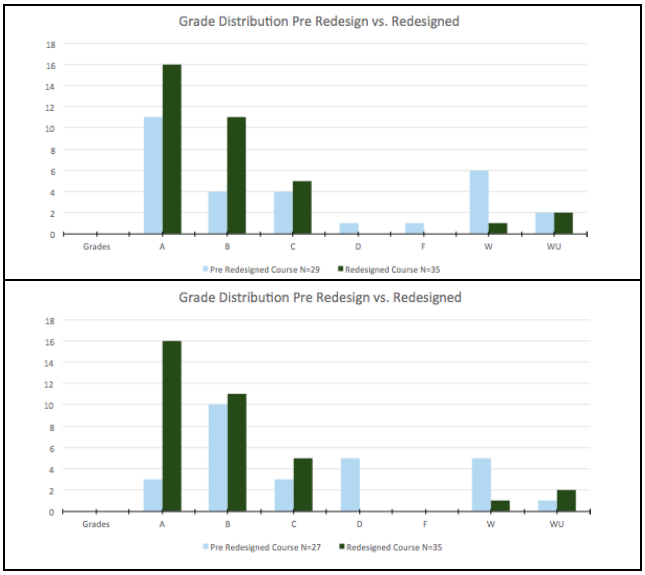

Fig. 4. Final Grade: a) Fall '14 vs. Spring '17; b) Spring '16 vs. Spring '17

ACKNOWLEDGMENT

We acknowledge that the funding for this work was partially supported by California State University Chancellor's Office for Course Redesign with Technology program (CRT grant), as well as by the National Science Foundation (NSF award number DUE-1626612).

REFERENCES

[1] P. Brusilovsky, "Adaptive and Intelligent Web-based Educational Systems." International Journal of Artificial Intelligence in Education, vol. 13 (2-4), pp. 159-172, 2003.

[2] E. Lavieri, "A Study of Adaptive Learning for Educational Game Design." ProQuest UMI, 2014.

[3] J. L. Bishop, J. Lowell, and M. A. Verleger, "The flipped classroom: A survey of the research." ASEE National Conference Proceedings, Atlanta, GA. Vol. 30, No. 9, 2013.

[4] D. Arendale, "History of supplemental instruction (SI): Mainstreaming of developmental education." Histories of developmental education, pp. 15-28, 2002.

[5] P. Blumenfeld, E. Soloway, R. Marx, J. Krajcik, M. Guzdial, and A. Palincsar, "Motivating project-based learning: Sustaining the doing, supporting the learning." Educational Psychologist, vol. 26 (3/4), pp. 369-398, 1991.

[6] A. Cockburn, and L. Williams, "The costs and benefits of pair programming in extreme programming examined." Boston, MA: Addison Wesley, 2001.

[7] Smart Sparrow, http://aelp.smartsparrow.com/

[8] A. Marcus-Quinn, and Y. Diggins. "Open educational resources." Procedia-Social and Behavioral Sciences 93, pp. 243-246, 2013.

[9] OpenDSA Project, http://iiscs.wssu.edu/drupal/OpenDSA

[10] OpenDSA CS3 Book, https://opendsa.cs.vt.edu/ODSA/Books/CS3/html/

[11] ZyBook, http://www.zybooks.com/