

The Impact of Course Transformation on Student Learning and Success in Fundamental Electrical Engineering/Computer Science Courses

Dr. David O. Johnson, University of Kansas

David O. Johnson is a Lecturer in the Electrical Engineering and Computer Science department at the University of Kansas in Lawrence, KS, USA. He received his BSEE and MSEE from Kansas State University and his PhD in Computer Science from the University of Kansas. Prior to two post-doctoral research appointments at the Eindhoven University of Technology in the Netherlands and in the Applied Linguistics Speech Laboratory at Northern Arizona University, Flagstaff, AZ, USA, he was an Adjunct Professor in the Computer Science Electrical Engineering department at the University of Missouri – Kansas City. Before beginning his academic career, he spent 31 years in industry as a manager and software developer and consultant.

Dr. Molly A. McVey, University of Kansas

Dr. Molly A. McVey is a post-doctoral teaching fellow at the University of Kansas School of Engineering where she works with faculty to incorporate evidence-based and student-centered teaching methods, and to research the impacts of changes made to teaching on student learning and success. Dr. McVey earned her Ph.D in Mechanical Engineering from the University of Kansas.

Christopher Patrick Melgares, University of Kansas

Graduate student at the University of Kansas

The Impact of Course Transformation on Student Learning and Success in Fundamental Electrical Engineering/Computer Science Courses

Introduction

Studies repeatedly show improvements in learning, achievement, and success for students after implementation of active learning and student-centered teaching practices. Active learning improves retention of content, achievement level, and success in courses [1, 2]. Research on flipped classrooms in engineering education has shown positive effects including increased retention, better performance on learning outcomes, and a high level of student satisfaction, especially with time [3]. Importantly, studies have also shown a differential improvement in performance for under-represented groups and previously lower-performing students [4, 5]. This paper describes the process of transforming a traditional lecture course first to a hybrid active learning/traditional lecture format and then to a fully flipped format. This paper then describes the impact of the course transformations on student learning and student success in a freshman level Electrical Engineering/Computer Science course, Introduction to Digital Logic Design.

Introduction to Digital Design is a fairly standard course included in most Electrical Engineering and Computer Science curricula. It is an introductory course in digital logic. Over the past two years, the instructor has changed the course from a traditional lecture to a hybrid active learning/traditional lecture format to a fully flipped format based on the SCALE-UP model [6] where students read, watch lecture videos, or review PowerPoint slides prior to class time and spend almost 100% of class time working. A previous study [7] found evidence of improved learning in a fully-flipped version of the course compared to the “active learning” version of the course. This paper seeks to broaden the scope of this work by investigating multiple semester’s worth of student performance data and by examining student success and particularly the success of women and under-represented minorities. The research questions are: 1) Does student performance on learning objectives improve in the fully flipped version of the course compared to the active learning version? 2) Is student success in the course improved in the active and/or flipped versions of the course compared to traditional lecture? 3) Is student success differentially impacted for under-represented groups? 4) What are student perceptions of the fully flipped course model?

Course Description

At this institution, Introduction to Digital Logic Design is an introductory course in digital logic circuits covering number representation, digital codes, Boolean Algebra, combinatorial logic design, sequential logic design, and programmable logic devices. It is a four credit hour course with a required lab and optional discussion section. It is offered in the Fall and Spring semesters (16 weeks per semester). Every Electrical Engineer (EE), Computer Engineer (CoE), and Computer Science (CS) major is required to complete the course and it is generally taken in the student’s freshman year. It is a gateway course to several other required courses in the department as illustrated in Figure 1. Between 110 and 180 students take the course each semester. The course includes honors students who are assigned additional homework and lab assignments.

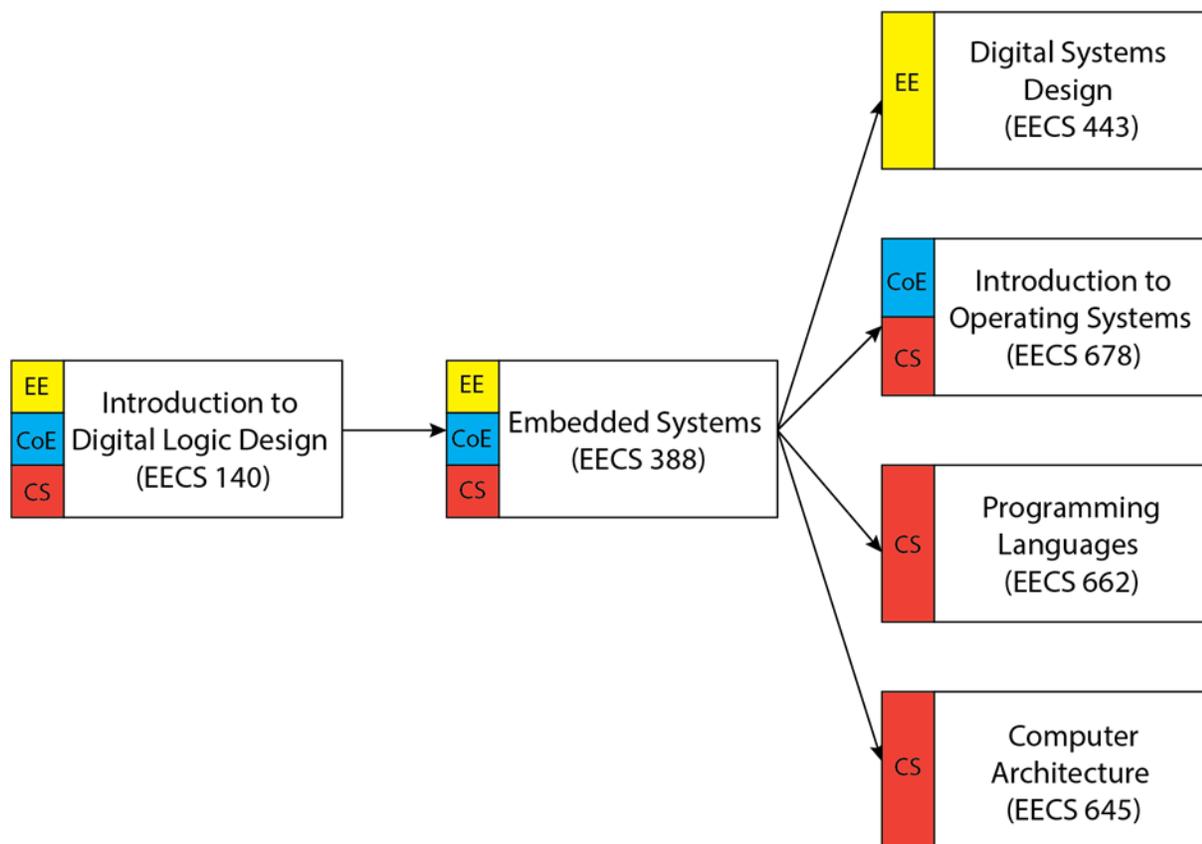


Figure 1. Curriculum map for digital logic design.

On completion of the course, the student is expected to be capable of: 1) representing a combinational logic function as a truth table, as a Boolean expression including various canonical forms, and as a logic circuit, and translating between these representations; 2) translating a simple logic problem expressed in prose to a combinational logic function; 3) simplifying a combinational logic function using K-maps and other techniques; 4) converting numbers between decimal and binary (and related) forms and designing simple digital circuits to perform numerical arithmetic functions; 5) designing combinational circuits using common building blocks; 6) designing flip-flop, register, and counter circuits; 7) implementing simple finite state machines from written specifications; and 8) writing VHDL code for simple digital circuits.

Introduction to Digital Logic Design has been taught at this institution for many years and is a typical introductory course in digital logic circuit design which is included in most engineering curricula. It has been taught in a variety of formats by a number of instructors. Generally, it has been taught by someone with a Ph.D. with graduate teaching assistants teaching the lab sections. In this paper, we focus on three semesters taught by a single instructor. The first semester was taught utilizing a blend of lecture and active learning activities. The second and third semesters were taught using the fully flipped format, with all information delivery occurring outside of class and active learning activities during the entirety of the class time. Each semester also had a lab component to the course, which was not impacted by the changes made to the “lecture” portion of the course. Each semester, the instructional team consisted of the main instructor, an

undergraduate assistant, and 2-4 “undergraduate teaching fellows,” who assisted with answering questions during in-class activities in addition to the graduate teaching assistants who taught the labs and graded the class work.

Fall 2017: Mixed lecture and active-learning

At the beginning of the semester, the students were given a schedule of the topics that would be covered each class period and the sections in the textbook corresponding to those topics. The students were not required to prepare for class as the lectures were designed to provide all the instruction the student would need to complete the homework and pass the exams. The PowerPoint slides for each lecture were posted on the University’s Learning Management System (LMS) approximately a week before the lecture for students to study.

Each section of the course was held in a traditional, lecture-style classroom or auditorium. During the 75 minute class periods, the instructor lectured for approximately 30 minutes. Previously posted PowerPoint slides were presented and annotated with an electronic stylus. The lecturer stopped after each major point and asked for questions. After the lecture, the students were given an in-class problem to complete and hand-in at the end of class. The students were encouraged to work together to solve the problem, but each student had to turn in their own in-class problem solution. The in-class problems were graded and comprised 10% of the student’s grade. The instructor and two undergraduate teaching assistants, who had been given the solutions, walked around the room helping students as they worked on the problem. A typical in-class problem was: *Design a circuit that multiplies an 8-bit unsigned number by 9, using only 1 full-adder.*

Each week the students were given 8-10 problems from the textbook to complete as homework. Honors students were given 2-3 additional problems, which might require further reading from the textbook. The homework assignments were posted at least a week before they were due and at the same time the lecture slides corresponding to the assignments were posted. An optional 2-hour help session was held during the discussion section the night before the homework assignments were due. The help session was led by an undergraduate teaching assistant. All of the homework assignments were graded and comprised 20% of the student’s grade. The solutions for the homework assignments were posted on-line shortly after the assignments were turned in. A typical homework problem might be: *(a) Show that the circuit in Figure P3.2 is functionally equivalent to the circuit in Figure P3.1. (b) How many transistors are needed to build this CMOS circuit?*

Three open-book, open-note non-comprehensive exams were given. The students were allowed to use their phones, tablets, or laptops to access their notes, lecture slides, and homework solutions. The first two exams were given during the 75 minute class periods and the last one was given during the 150 minute final exam period. The last exam was approximately twice as long and covered twice as many topics as the first two. Honors students were given the same exam as the non-honors students. The exams were equally weighted (including the longer final exam) and represented 45% of the student’s grade.

During the class period prior to the exam, the students were given the number of questions and points per question along with the learning objective that the question covered and the section of the book where the learning objective was covered. For example: *Create a truth table for a logic function – Section 2.3 (6 points)*. An exam review was also conducted by the undergraduate teaching assistant during the help session the night before the exam. A typical exam question might be: *Create a truth table for the logic function: $f = x(x \cdot y)(y + z)$* .

Spring and Fall 2018: Fully Flipped Classroom

The second and third semesters were taught using a flipped classroom model. The Spring 2018 semester was conducted in a classroom specifically designed for active learning. Students sat in 5 person U-shaped tables, and the instructor and assistants were easily able to move around and assist students as they worked (Figure 2, left). The Fall 2018 semester was conducted in a large lecture-style auditorium, which was somewhat designed for active learning (Figure 2, right). The students sat at long tables on elevated platforms, which facilitated them working together on problems, but made it somewhat difficult for the instructor and assistants to easily move around and help the students as they worked.



Figure 2. Photo of flipped classroom implementation. Spring 2018 (left). Fall 2018 (right).

At the beginning of the semester, the students were given a schedule of the modules to be covered each class period. Students were responsible for reviewing the material and taking an online quiz for each module prior to class. They could review the material by listening to the instructor in pre-recorded lectures, reading the book, going over the slides, looking it up on the internet, or all of these. During the class period, they worked on in-class problems.

In the Spring 2018 semester, the two sections of the class were held on different schedules. One section was three 50-minute class periods on Monday, Wednesday, and Friday (MWF); the second section was two 75-minute class periods on Tuesday and Thursday (TuTh). Because of the differences in the length of the lectures, the MWF students did two in-class problems each class period, while the TuTh students did three in-class problems. So, six learning objectives were covered each week in both sections. The pre-record videos, slides, and on-line quizzes were broken down by learning objective, so the MWF students had to prepare for two of them before each class and the TuTh students had to prepare for three before each class. In the Fall 2018

semester, both sections were held on TuTh and followed the same format as the Spring 2018 TuTh section.

The learning objectives for the pre-flipped first semester (Fall 2017) were covered in 28 lectures which were each approximately 30 minutes in length. The goal of the flipped-class was to divide these lectures into 84 modules each representing approximately 10 minutes of the first semester lectures. The 84 learning objectives for the flipped class are shown in Table 1.

Table 1. Flipped class learning objectives.

No.	Description	No.	Description
1	Give the truth table for the AND and OR functions.	43	Design a carry-lookahead adder.
2	Give the truth table for the NOT of a logic function.	44	Design a hierarchical carry-lookahead adder.
3	Create a truth table for a logic function.	45	Design an array multiplier for unsigned binary numbers.
4	Draw the logic network of gates that implements a logic function.	46	Multiply signed binary numbers with 2's complement arithmetic.
5	Use Boolean Algebra to reduce a logic function.	47	Convert a fixed-point binary number to decimal.
6	Prove a Boolean identity with a Venn Diagram.	48	Give the decimal exponent range and precision of a single- or double-precision IEEE floating point number.
7	Give the canonical sum-of-products (SOP) for a logic function.	49	Design an n-to-1 multiplexer.
8	Give the canonical product-of-sums (POS) for a logic function.	50	Design a switch with a multiplexer.
9	Determine whether to use a canonical sum-of-products (SOP) or product-of-sums (POS) to implement a logic function.	51	Synthesize a logic function with a multiplexer.
10	Give the truth table for the NAND and NOR functions.	52	Factor a single variable out of a Boolean expression with Shannon's Expansion Theorem.
11	Draw the logic network of only NAND gates that implements a logic function.	53	Apply Shannon's Expansion Theorem to a single variable of a logic function to implement it with a multiplexer.
12	Draw the logic network of only NOR gates that implements a logic function.	54	Apply Shannon's Expansion Theorem to multiple variables of a logic function to implement it with a multiplexer.
13	Give the truth table for the XOR function.	55	Design a binary decoder.
14	Give the differences between a positive logic and a negative logic implementation using voltage levels.	56	Design a binary encoder.
15	Know which voltage (high or low) applied to a NMOS transistor opens or closes the transistor switch.	57	Design a priority encoder.
16	Know which voltage (high or low) applied to a PMOS transistor opens or closes the transistor switch.	58	Design a comparator for two unsigned numbers.
17	Draw the NMOS realization of a logic network.	59	Design a comparator for two signed numbers.
18	Draw the CMOS realization of a logic network.	60	Design a basic SR latch with NOR and NAND gates.
19	Use a Karnaugh map to find the minimum-cost sum-of-products (SOP) for a 2-variable logic function.	61	Design a gated SR latch with NAND gates.
20	Use a Karnaugh map to find the minimum-cost sum-of-products (SOP) for a 3-variable logic function.	62	Design a gated D latch.
21	Use a Karnaugh map to find the minimum-cost sum-of-products (SOP) for a 4 and 5 variable logic function.	63	Design a master-slave D flip-flop with two gated D latches.
22	Identify the literals, implicants, prime implicants, cover, and cost of a logic function.	64	Design a positive edge triggered D flip-flop from NAND gates.
23	Know how to use essential and non-essential prime implicants to find a minimum cost cover for a sum-of-products (SOP) implementation.	65	Design a negative edge triggered D flip-flop from NOR gates.
24	Know how to use essential and non-essential prime implicants to find a minimum cost cover for a product-of-sums (POS) implementation.	66	Describe how an asynchronous and synchronous D flip-flop clear and preset work.
25	Give the canonical SOP for a K-map with don't cares.	67	Explain the differences between types of flip-flops and latches.
26	Given two logical functions, draw a logical circuit with multiple output circuit sharing.	68	Design a shift register with D flip-flops.
27	Explain what fan-in means, why high fan-in is a problem, and the engineering trade-offs of avoiding high fan-in.	69	Design a parallel-access shift register with D flip-flops.
28	Simplify a logical function with factoring.	70	Design an asynchronous up-counter or down-counter.
29	Simplify a logical function with disjoint functional decomposition.	71	Design a synchronous up-counter or down-counter.
30	Simplify a logical function with non-disjoint functional decomposition.	72	Design a modulo-n counter with a synchronous reset.
31	Convert a multilevel circuit to NAND or NOR gates.	73	Create a state diagram for a Moore-type finite state machine (FSM).
32	Trace a multi-level circuit to determine its logical function.	74	Derive the state assigned table for a Moore-type finite state machine (FSM).
33	Convert a binary number to decimal.	75	Implement the digital logic circuit for a Moore-type finite state machine (FSM).
34	Convert a binary number to octal and hexadecimal.	76	Demonstrate how different state assignments can have an effect on the cost of a finite state machine (FSM).
35	Design a full adder (FA).	77	Simplify the logical expressions of a finite state machine (FSM) with one-hot encoding.
36	Design an n-bit ripple-carry adder.	78	Design a Mealy-type finite state machine (FSM).
37	Represent a signed binary number in 2's complement form.	79	Design a Mealy-type finite state machine (FSM) serial adder.
38	Add signed binary numbers with 2's complement arithmetic.	80	Design a Moore-type finite state machine (FSM) serial adder.
39	Subtract signed binary numbers with 2's complement arithmetic.	81	Design a finite state machine (FSM) counter with D flip-flops.
40	Design an adder/subtractor unit.	82	Design a finite state machine (FSM) counter with JK flip-flops.
41	Design an adder/subtractor unit that detects arithmetic overflow.	83	Design a finite state machine (FSM) that counts non-sequential pulses on a line.
42	Calculate the critical-path delay for a multi-level circuit.	84	Analyze the behavior of an existing finite state machine (FSM).

The pre-recorded lectures were prepared by recording the PowerPoint slides for each module with a voice over and electronic annotation by the instructor and available online. A pre-class quiz for each module, designed to test the student's comprehension of the module's learning objective, were available on-line and auto-graded. The quizzes were primarily fill-in the blank and usually required the student to design digital logic circuits off-line and then enter the answers on-line. Figure 3 is an example of one of the quizzes. The quizzes represented 25% of

the student's grade and had to be completed prior to the beginning of class. The instructional team felt that most students did the pre-class work and came to class prepared, likely due to the high weight of the quizzes on their course grade as well as the fact that preparing allowed them to complete the in-class work, which was also graded.

Question Referring to the basic SR latch constructed with NOR gates as depicted in Figure 7.5(a) and Slide 4 of Module 60, fill-in the following table:

Time	S	R	Q _a	Q _b
t1	0	1	[Qa1]	[Qb1]
t2	0	0	[Qa2]	[Qb2]
t3	1	0	[Qa3]	[Qb3]
t4	0	0	[Qa4]	[Qb4]
t5	0	0	[Qa5]	[Qb5]

Figure 3. Pre-class quiz example.

Each module had an in-class problem. The in-class problems were designed to take the average student 25 minutes to complete. The students did not see the problems until class started and were required to turn them in at the end of class for a grade, which comprised 25% of their grade. An example of an in-class problem is: *Design an n-bit inverter. The circuit will have n number of inputs, x_n through x_1 , and n output bits, f_n through f_1 . The circuit will have an additional input, s. $f_k = x_k$ (not inverted) when $s=0$ and $f_k = !x_k$ (inverted) when $s=1$, where $1 \leq k \leq n$. Use only XOR gates in your implementation.*

The students were encouraged to work together to solve the problems, but each student had to turn in their own in-class problem. Students tended to work in groups of 2-4, although some students preferred to work alone. The instructor and undergraduate teaching assistants walked around the room answering questions. These activities took the place of homework, which was not assigned in the class. In-class problems were collected and graded. Occasionally, the instructor might spend 1-2 minutes giving announcements, but usually the students began working on the problems and asking questions as soon as they were displayed on the projector screens, often 5-10 minutes before class officially started.

Four open-book, open-note non-comprehensive exams were given. The students were allowed to use their phones, tablets, or laptops to access their notes on the in-class problems and pre-class quizzes, lecture slides, and lecture videos. The first three exams were given during the 120-minute discussion periods and covered Modules 1-20, 21-38, and 39-56, respectively. The last exam was given during the class's 150-minute final exam period and covered Modules 57-84. The exams were equally weighted and together represented 25% of the student's grade. An exam

review was also conducted by an undergraduate teaching assistant a week before each exam was given. In order to track student learning, similar (but not the same) questions to those in the first semester, pre-flipped class, were asked.

Methods

Classroom Activity

In order to compare classroom activities between semesters, we utilized the Classroom Observation Protocol for Undergraduate STEM (COPUS) [5]. To perform the COPUS, a trained observer visited each course three times in a two-week period. For each COPUS observation, the observers indicated in each two-minute interval of class time whether or not 13 student and 12 instructor behaviors (listed in Table 2) occurred. Data from the three observations of each course were averaged, and then the thirteen behaviors were collapsed into four categories. For students, those categories were *receiving*, *working* (included individual thinking/working, clicker question in groups, working in groups, and other group), *talking to class* (answering questions, asking questions, whole class discussion, and student presentation), and *other* (waiting, other). For instructors, those categories were *presenting* (lecturing, real-time writing, and demo/video), *guiding* (follow-up, posing question, clicker question, answering question, moving around to groups, and one-on-one), and *other* (waiting, other). The undergraduate teaching assistants were treated as instructors in the study.

Table 2. Student and instructor behaviors tracked on COPUS observation

COPUS Student Behaviors	COPUS Instructor Behaviors
Listening [L]	Lecturing [Lec]
Answering Questions [AnQ]	Real-time writing [RtW]
Asking Questions [SQ]	Demo/Video [D/V]
Whole class discussion [WC]	Follow-up [Fup]
Student Presentation [SP]	Posing question [PQ]
Individual thinking/working [Ind]	Clicker question [CQ]
Clicker question in groups [CG]	Answering Question [AnQ]
Working in groups [WG]	Moving around to groups [MG]
Other group [OG]	One-on-One [1o1]
Predicting the outcome of something [Prd]	Administrative [Adm]
Test/Quiz [TQ]	Waiting [W]
Waiting [W]	Other [O]
Other [O]	

Student Performance: To quantify student performance, student achievement on common learning objectives across the courses and “Drop, Fail, or Withdraw” (DFW) rates were analyzed. Performance on exam questions mapped to the same learning objective were compared between the hybrid active learning semester (Fall 2017) and two semesters of the fully-flipped course (Spring 2018 and Fall 2018). The Independent Samples *t*-test was used to test for significant differences in performance across semesters with $p < .05$ used for significance. IBM SPSS Statistics software was used to conduct all statistical analysis.

DFW rates were compared between an average of 8 semesters (Fall 2013-Spring 2017) when the course was taught using traditional lecture, the hybrid active learning semester, and one of the

fully flipped semesters. Additionally, DFW rates for females, males, under-represented minorities (American Indian, Black, Hawaiian/Pacific Islander, Hispanic, and those of 2 or more races), and international students were compared. It is important to note that there were also changes in instructors across the compared semesters (there were several different instructors over the 8 semesters of traditional lecture, and the change to active learning coincided with a change in instructor, who was the same for the fully flipped semester). Although the Fall 2018 data for the fully flipped class was included in the other analyses, it was not included in the DFW analysis because the DFW rate for the semester was not available at the time of writing this paper.

Student Perception: Surveys were given in order to understand student opinions about the operation of the flipped classroom model. The survey was administered online using Qualtrics. The survey was not anonymous, because students were given extra credit for completing it. However students were informed that the results of the survey would only be seen by the post-doctoral teaching fellow assisting with this project, and the instructor of the course would only see the aggregated results. Surveys asked students Likert-scale questions about the effectiveness/usefulness of each course component, and how they prepared for class.

Results and Discussion

Classroom Activity: The COPUS observations confirmed the fully-flipped nature of the Spring 2018 and Fall 2018 semesters. Almost the entire class time was spent “working” for students and “guiding” for instructors (Figure 4).

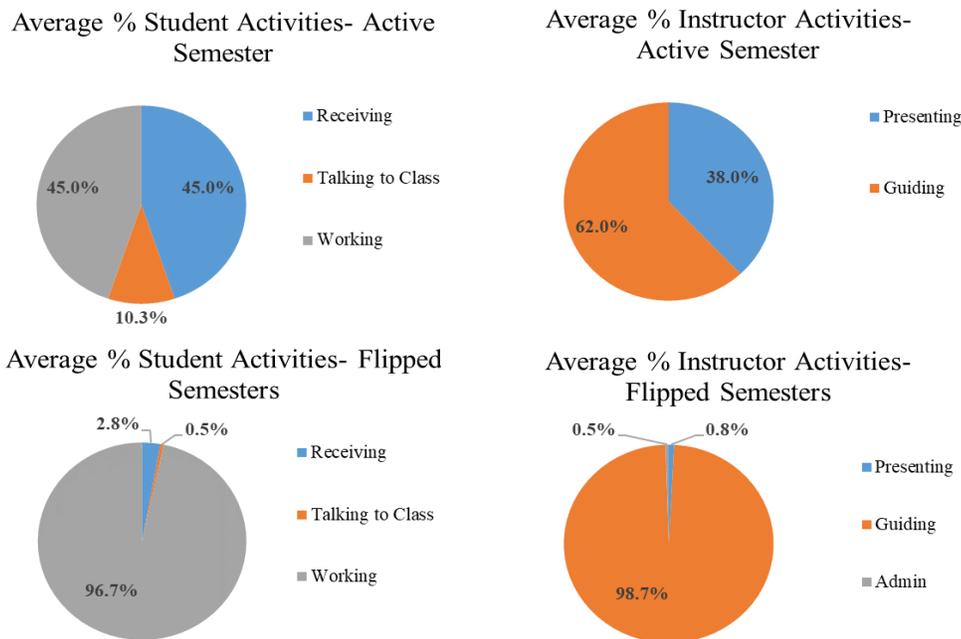


Figure 4. COPUS results for Active compared to Flipped Semesters.

Student Performance: Learning Objective Achievement

Of the 58 learning objectives compared, students in the fully flipped version of the course scored significantly better on 22 learning objectives (37.9%). There was no significant change in performance on 30/58 learning objectives (51.7%), and a significant decline in performance in 6/58 objectives (10.3%). Following are details on each exam.

Exam 1: Results of performance on Exam 1 by learning objective are shown in Figure 5. On the first exam there was a significant improvement in the flipped semesters in 9/14 (64.3%) of learning objectives. There was no significant change in 5/14 (35.7%) of objectives. The largest improvements were in learning objectives 17 (Draw the NMOS realization of a logic network) and 18 (Draw the CMOS realization of a logic network). These modules along with 14, 15, and 16 (where there was also improvement) are the only ones that cover the actual electronics of digital logic design. This is the part of the course that is most difficult for Computer Science majors, who make up the majority of the class. During the Fall semester of the fully flipped course, the instructional team noticed that the Computer Science students were really struggling with the in-class problems for these learning objectives and concentrated on helping them more in both the Fall and Spring semesters of the flipped course. This probably was a major factor in the significant improvements.

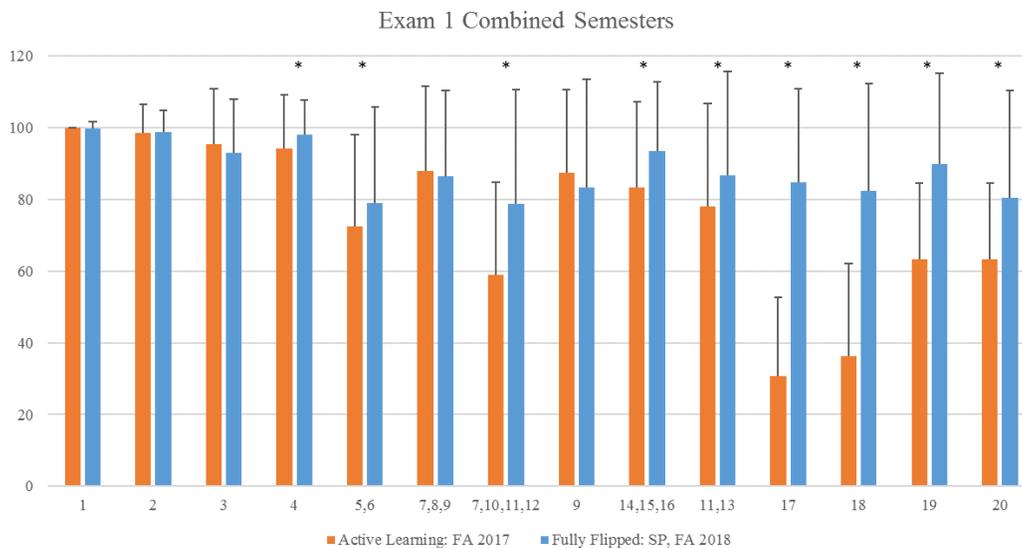


Figure 5. Learning objective performance on Exam 1 in active learning (orange) compared to fully-flipped (blue bars) semesters. *Indicates statistical significance by t-test.

Exam 2: Results of performance on Exam 2 by learning objective are shown in Figure 6. There was a significant improvement in the flipped semesters in 3/13 (23.1%) of the objectives. There was no change in 8/13 (61.5%) of objectives, and a significant decline in 2/13 (15.4%) of the objectives. The improvement in Modules 21, 22, and 23 was a continuation of the improvement from Exam 1 in Modules 19 and 20. These 5 modules cover the use of Karnaugh maps to simplify logic expressions, which is one of the major building blocks of digital logic design. Students who do not grasp this concept early in the course will have trouble later on. Like the improvements in learning objectives 14-18, improvements in learning objectives 19-23 were most likely due to extra effort put in by the instructional team during the Spring semester of the

flipped class because they noticed the students struggling with it during in-class problem solving in the Fall semester of the flipped class.

Performance on objectives 29, 30, 33, and 37 declined significantly. Learning objectives 29 and 30 cover the rather arcane and difficult concept of functional decomposition. Understanding functional decomposition is more important than understanding the rather esoteric difference between disjoint and non-disjoint functional decomposition. These two modules should be revised to focus more on how to perform functional decomposition than the difference between disjoint and non-disjoint functional decomposition. The decline in learning objectives 33 and 37 is interesting. The exam question for these objectives asks the students to convert a decimal number to a 2's complement representation of the number. 2's complement is how computers add and subtract numbers. It is a very basic concept in digital logic design. The interesting part is that the exam questions for objectives 38 and 39 ask the students to perform 2's complement addition and subtraction. There was no difference in their performance on addition and a significant improvement in performance on subtraction (see Exam 3 module 39). So, it appears the students understand how to perform 2's complement arithmetic, but not how to convert a signed decimal number to its 2's complement form (Modules 33 and 37). A thorough review of the material on Modules 33 and 37 revealed that it presented two alternate forms of representing decimal numbers in binary, signed and 1's complement and then showed why these were not as good as 2's complement in digital logic design. In retrospect, this was likely confusing to the students. A better alternative would be to explain how to represent a number in 2's complement to avoid confusing the students with the signed and 1's complement formats. Once again the student's confusion was discovered as the instructional team helped the students work on in-class problems dealing with the conversion of signed decimal numbers to 2's complement form.

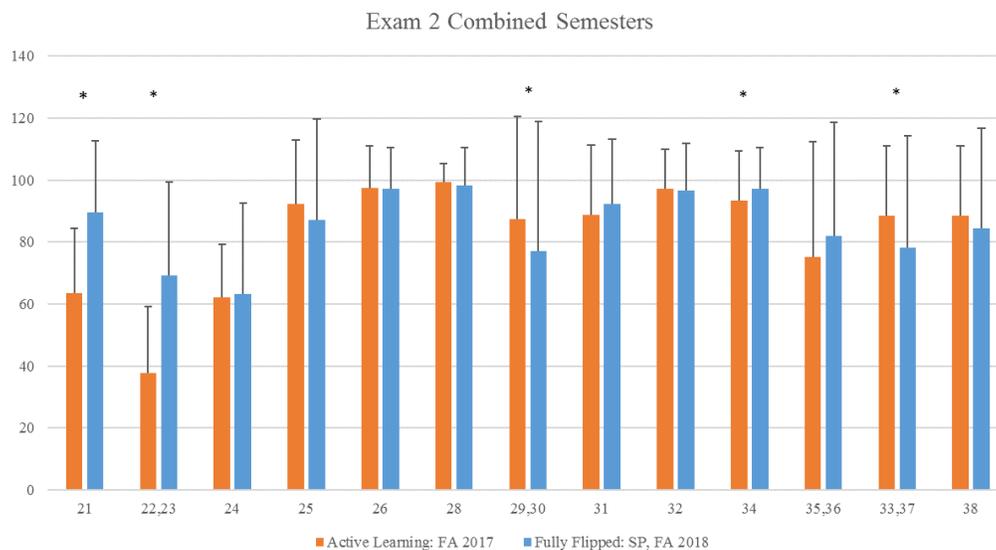


Figure 6. Learning objective performance on Exam 2 in active learning (orange) compared to fully-flipped (blue bars) semesters. *Indicates statistical significance by t-test.

Exam 3: Results of performance on Exam 3 by learning objective are shown in Figure 7. On the third exam, there was a significant improvement in 8/13 (61.5%) of the objectives and no change in 5/13 (38.5%) of objectives.

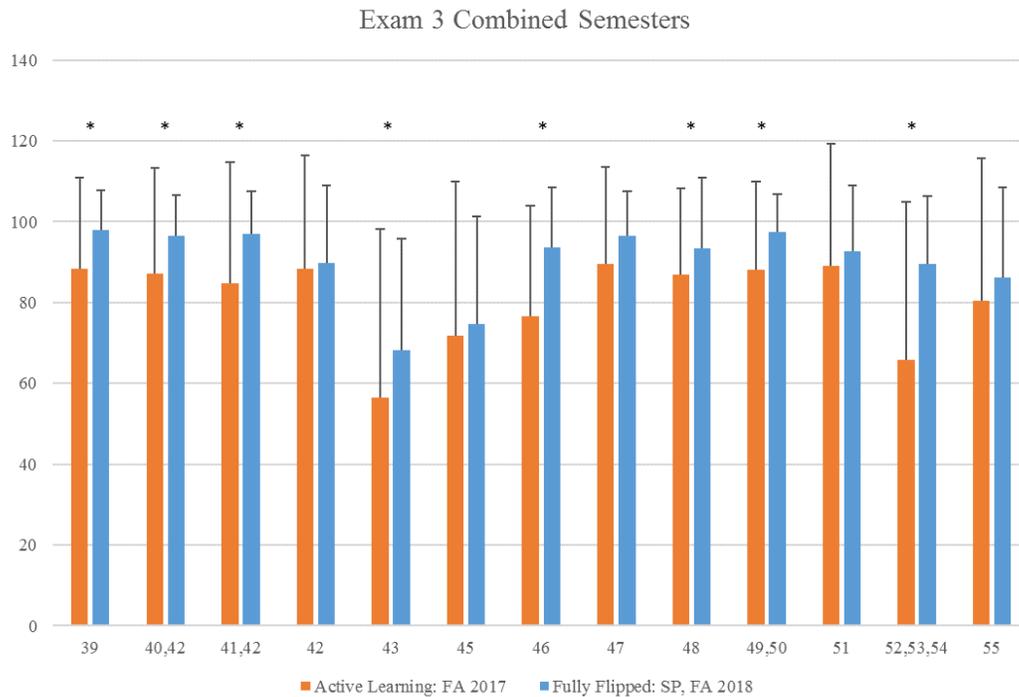


Figure 7. Learning objective performance on Exam 3 in active learning (orange) compared to fully-flipped (blue bars) semesters. *Indicates statistical significance by t-test.

Exam 4: Results of performance on Exam 4 by learning objective are shown in Figure 8. On Exam 4, given as the final exam, there was a significant improvement in 2/18 (11.1%) of objectives, a significant decline in 4/18 (22.2%) of objectives, and no change in 12/18 (66.7%) of objectives. Regarding the declines, the questions for Modules 63, 64, 65, and 67 required the students to understand the differences between latches and flip-flops. The declines indicate a need for improvement in the material for these objectives. The declines in objectives 78, 79, and 84 (on finite-state machines) may have had more to do with lack of motivation on the students' part, than with the course material since the comprehension of finite-state machines was tested multiple times in the last six questions of the exam. By lack of motivation, we mean the students in the flipped class room knew the final exam was only 6.25% of their final grade while it was 15% of the active learning's students' final grade. Several students in the flipped semesters told us they were not going to take the final exam because it wouldn't change their grade. A number of exams were turned in with only the easiest problems done. A spot check indicated the final exam would not change these student's final grade. Of the students where it did matter, the answers were correct. So, part of what we see here is an effect of combining the correctly answered questions with those that were not answered at all.

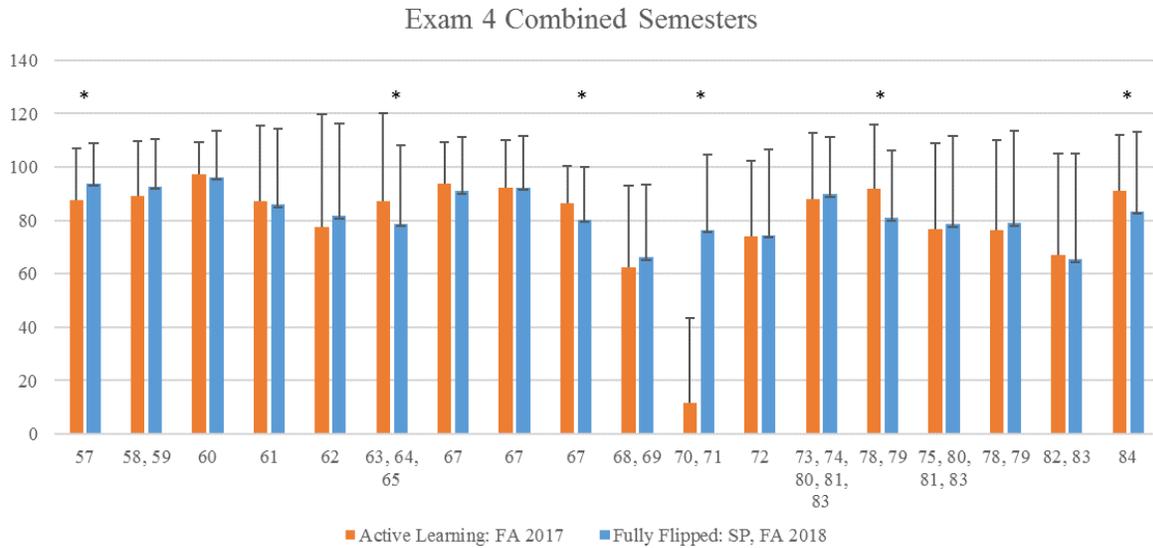


Figure 8. Learning objective performance on Exam 4 in active learning (orange) compared to fully-flipped (blue bars) semesters. *Indicates statistical significance by t-test.

Student Performance: DFW Rates. Overall, success in the course (defined as earning an A, B, or C in the course) improved for all students in the flipped classroom pedagogy compared to the traditional lecture, and in some groups the flipped classroom improved success rates in the class compared to the active learning pedagogy. The overall DFW (D/F/withdraw) rate was 18.0% for the average of 8 semesters taught in a traditional format. The DFW rate in the traditional semesters was even higher for females (23.4%) and under-represented minorities (URM) (21.3%). The DFW rate for the flipped version of the course was reduced by more than half to only 7.7% for females. The DFW rate for URM was reduced to 14.3% in the flipped version of the course. All International students were successful (DFW rate = 0%) in the flipped version of the course. DFW rates for each semester and group are shown in Figure 9.

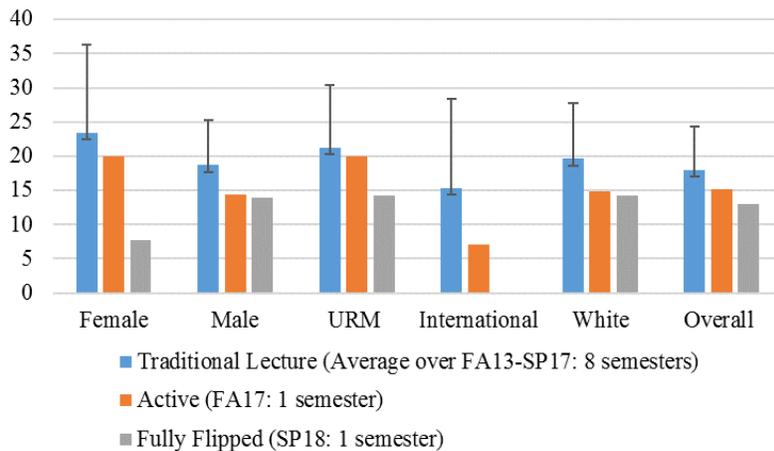


Figure 9. % DFW rates in traditional (blue), active (orange), and flipped (grey) versions of the course by demographic group.

Student Perception: Student survey data from 226 respondents revealed a high level of satisfaction in the flipped version of the course. Figure 10 shows survey responses to several questions regarding their preparedness and satisfaction with different aspects of the course. The students overwhelmingly reported coming to class prepared to work the in-class problems (93.8% *agreed* or *strongly agreed*). The video lectures, PowerPoint slides, and online quizzes were useful to a majority of students. Most striking was that 92.9% of respondents *strongly agreed* (66.4%) or *agreed* (26.6%) that the in-class problems were useful, and 86.7% *strongly agreed* (63.3%) or *agreed* (23.5%) that the class meeting was useful to their learning. A majority of respondents (82.3%) *strongly agreed* or *agreed* that they had adequate support for learning in the course.

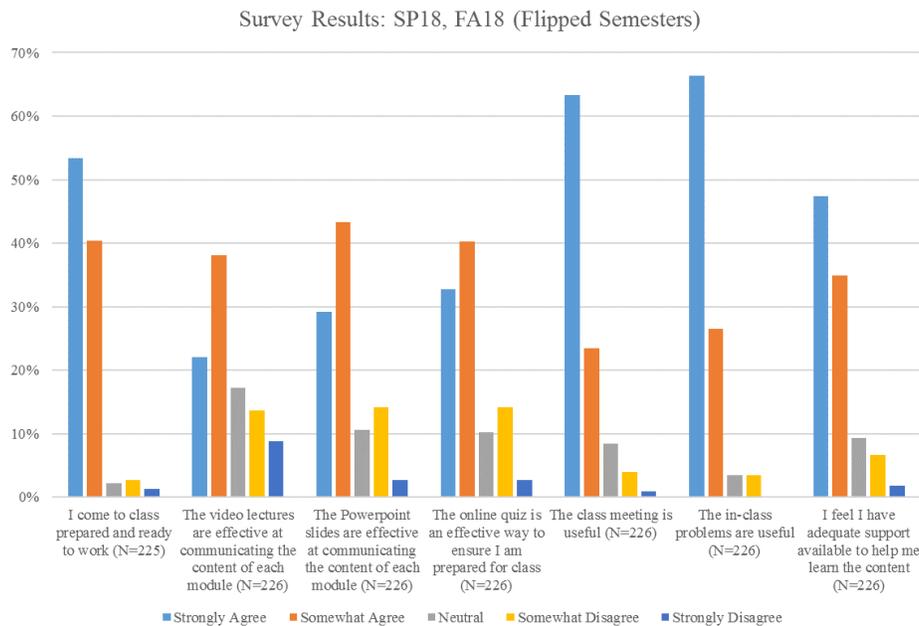


Figure 10. Student survey results from fully flipped semesters.

Discussion

The results of this study show 1) an overall improvement in student learning in a flipped compared to a hybrid lecture/active learning course taught by the same instructor, 2) improved rates of course success (A,B,C) for all students, and 3) a dramatic improvement in rates of course success for female and under-represented minorities. This study adds to the body of work supporting the effectiveness of the flipped format for teaching STEM, and specifically electrical engineering courses [4, 8, 9]. However, implementation is key- and not every flipped classroom is the same. Here we will describe the insights gained and lessons learned from this project.

In the fully flipped classroom, the instructor role changed considerably. In the hybrid active learning model, the instructor was primarily passive. The instructor stood at the front of the classroom waiting for students to approach him with questions. In the fully flipped model, the instructor spent every minute of class conversing with students and answering questions while walking from group to group. The instructor was better able to understand student challenges and the ways students engage with the course material, which allowed him to tailor class materials

such that students come to class prepared to work on the in-class problems. In the fully-flipped model it is easy to get a sense of the level of difficulty of the problem sets, the processes that students undertake when trying to solve an unfamiliar problem, and common misunderstandings. This connection to students and their learning is motivating as an instructor.

Another important role in this model is the instructional team including graduate and undergraduate teaching assistants. The role of the undergraduate teaching assistants is to roam the room answering questions. These students are undergraduates who recently were successful (A or B) in the course. They are instrumental in being able to facilitate this model with a large classroom. These students are given several training sessions through the School of Engineering during the semester focusing on the art of asking questions and guiding students in their learning. The role of the GTAs is to grade the in-class work, as it is important that students are accountable for the work done in-class

While implementing a fully-flipped course required a lot of initial preparation work (e.g., designing problem set modules, creating question banks for Blackboard quizzes, and preparing video lectures and narrated PowerPoint slides), the instructor thinks that the effort was worth it. The results presented here suggest that students are more engaged and show improved achievements of course learning goals. The instructional team is also more satisfied. Finally, the authors suggest the following when considering implementing a flipped classroom design:

- Make the classroom space work for how you want to teach. While a setup of group tables with plenty of space for facilitators to walk around seems ideal, even an auditorium-style classroom has worked well for our flipped classroom.
- Grading in-class problems takes about as much time as grading homework problems. Prioritizing in-class learning experiences for students will not necessarily correlate with more time grading.
- The time investment needed to flip a classroom is significant, but the majority of this work will not need to be re-done every semester. For example, designing question banks for online quizzes was very time consuming, but now the instructor can continue to pull from these questions, and all quiz grading is done automatically, online, which saves time in the long run. In addition, all of these materials may be utilized by other instructors of the course who may want to teach in this format, making this model more sustainable.

Conclusion

This study highlights the advantages of teaching in a flipped format where students prepare prior to class and spend the entire class time working. Based on the literature, this is exactly what one would expect. However this study shows that even compared to a hybrid active learning model (50% lecture, 50% working) with the same instructor and same content, there were significant improvements in learning objective achievement and course success in the fully flipped format, especially for under-represented groups. Additionally, student perception of the flipped format was very positive.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Number DUE1525775. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

- [1] R. M. Felder, D. R. Woods, J. E. Stice, and A. Rugarcia, "The future of engineering education II. Teaching methods that work," *Chemical Engineering Education*, vol. 34, pp. 26-39, 2000.
- [2] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, *et al.*, "Active learning increases student performance in science, engineering, and mathematics," *Proceedings of the National Academy of Sciences*, vol. 111, pp. 8410-8415, 2014.
- [3] B. Kerr, "The flipped classroom in engineering education: A survey of the research," in *2015 International Conference on Interactive Collaborative Learning (ICL)*, 2015, pp. 815-818.
- [4] Q.-V. Dang and D. D. Gajski, "Bringing in-class online-A hybrid solution," in *Fourth Interdisciplinary Engineering Design Education Conference*, 2014, pp. 12-17.
- [5] J. Roberts, A. Marshall, N. McLean, G. Baker, and A. Möller, "Demonstrating the impact of classroom transformation on STEM inequality in DFW rates ("D" or "F" grade or withdraw) for first-time freshmen, females, and underrepresented minorities through a decadal study of introductory geology courses," *Journal of Geoscience Education*.
- [6] R. J. Beichner and J. M. Saul, "Introduction to the SCALE-UP (student-centered activities for large enrollment undergraduate programs) project," *Proceedings of the International School of Physics "Enrico Fermi," Varenna, Italy*, 2003.
- [7] D. O. Johnson and M. A. McVey, "Implementation of a Flipped Classroom Model in Digital Logic Design," in *Midwest Section Meeting of ASEE*, Kansas City, MO, 2018, pp. 1-13.
- [8] G. J. Kim, E. E. Patrick, R. Srivastava, and M. E. Law, "Perspective on flipping circuits I," *IEEE Transactions on Education*, vol. 57, pp. 188-192, 2014.
- [9] K. Yelamathi and E. Drake, "A flipped first-year digital circuits course for engineering and technology students," *IEEE Transactions on Education*, vol. 58, pp. 179-186, 2015.