International Journal of Engineering Education Vol. 34, No. 6, pp. 1829–1837, 2018
Printed in Great Britain
0949-149X/91 $3.00+0.00

Improving Student Success Through an Effective Learner-Centered Course in Introductory Engineering, Mathematics, and Programming*

KUMAR YELAMARTHI
School of Engineering and Technology, Central Michigan University, Mt Pleasant, MI, USA. E-mail: k.yelamarthi@ieee.org

Academic success of engineering students in sophomore and junior years have been tied to their successful navigation of the first-year program that typically includes fundamental courses in engineering, programming, and mathematics. While mathematics is a core of engineering, it is often cited as a reason for poor performance in sophomore courses. Addressing this challenge, this paper reports a first-year course to bridge the student knowledge gap between engineering, programming, and mathematics, and demonstrates improved student learning. This new course was designed and taught for two years to 233 students. Student performance in the proposed course and follow-up sophomore circuit analysis course is tracked and analyzed. The effectiveness of the proposed course is validated through an increase in student performance and their perceptions of their learning experience across several academic years. Results showed that students have a better understanding of engineering and programming concepts as evidenced by their performance in the proposed first-year course, and in the follow-up sophomore circuit analysis course. In comparison with control and experimental groups, the number of students who are at risk of failure in Circuit Analysis course has decreased from 32% to 25%. Additionally, the number of students that have the potential to succeed but needs motivation has increased from 17.6% to 27%.

Keywords: active learning; first-year; problem-based-learning; programming efficacy; student perception

1. Introduction

The need to prepare future engineers for the rapidly evolving opportunities has never been greater. Identifying this, numerous researchers have presented ways to recruit a higher number of students in the engineering programs and retain their interest through the design of innovative and engaging first-year courses. Additionally, engineering educators from multiple disciplines have incorporated a broad range of instructional strategies such as problem-based learning (PBL), project-based learning, team-based learning, flipped and/or inverted learning in their teaching methodologies. However, recent research has shown that teaching sophomore introductory core subjects such as circuit analysis and computer programming have been a significant challenge for educators [1]. This lack of student performance and success has been primarily due to their lack of preparation in math concepts, and their ability to connect math to engineering concepts.

Mathematics courses have been predominately taught during the first two years of the engineering curriculum, beginning with calculus. However, majority engineering students do not take core engineering courses such as circuit analysis or statics until the sophomore year. Traditionally mathematics concepts are not taught with an emphasis on engineering applications, or how mathematics could be utilized to solve engineering problems. Research shows that as engineering heavily relies on math concepts, the students' mathematics self-efficacy influences their decision on persistence [2], and thereby their retention and success in the engineering programs.

In the current technology-rich era, students anticipate solving problems through simplified and automated methods using programming, with minimal interest to solve using traditional methods. They also expect their professors to be more engaging by demonstrating the reasons to learn a new concept and would like to immediately apply the concept to evaluate its respective benefits. Not doing so often results in student's loss of interest, inability to integrate themselves into engineering programs, inadequate performance in the core engineering courses, and limited persistence to remain in the engineering program.

Typically, programming courses are not taught in the first semester of the engineering programs as students need to master problem-solving skills, understand mathematical models, write pseudo-codes, learn semantics and syntax, which are obtained through developing cognitive abilities from engineering courses [3]. As learning programming as a time-consuming process that requires commitment and structured learning, it is not uncommon to find student unmotivated and leaving computer science related programs [4, 5].

Extensive research has been conducted on imple-

* Accepted 5 June 2018.

menting a broad range of pedagogical methods to teach programming [6]. Faux [7] has shown that students perform better in core engineering courses when they have completed a programming course with emphasis on problem-solving, pseudocode generation, and writing small programs. A recent survey has supported this by creating a first-year programming in many engineering programs [8]. It has been shown that educating students in fundamentals of computer programming combined with pedagogical approaches during the first-year increases pass rate in comparison with other methods [6, 9].

Comprehensively, engineering instructors need to continually adapt to the technological trend, use proven and new instructional methodologies in their respective courses. Currently, active-learning is a popular instructional methodology used in a collaborative and technology-rich, small-group setting [10]. When implemented effectively, this method has demonstrated higher student motivation, performance, and learning [9]. Simply using inductive methodologies will not lead to better student performance [11]. For PBL to be effective, educators need to demonstrate how to solve the problem with an emphasis on the process and application, step back and let students interact as a group and learn, and reinforce learning by offering support as necessary, and affirming the student understanding. O'Connell [12] has shown that this active learning method can help students with the development of a deeper conceptual understanding, problem-solving skills, nurture their self-directed independent learning abilities [13].

Addressing the stated challenges, combining the various pedagogical methods in the literature, this paper presents an active learning based first-year engineering-mathematics-programming (EMP) bridge course for engineering and engineering technology (ET) students. With a pre-requisite of just college algebra, this course addresses the salient math and programming topics used in a variety of core ET classes including traditional physics, electric circuits, statics, dynamics, and electronics through numerous design activities.

This paper is structured as follows. Section 2 describes the new course including course learning objectives, syllabus, activities. Section 3 presents the measures used, student perceptions, and student evaluations from two academic years. Section 4 presents a discussion on the findings, and Section 5 draws conclusions.

2. Course design and implementation

Literature has been published to demonstrate that students learn better through active learning strategies over conventional teaching methods [14–16]. Specifically, engineering concepts must be taught in a collaborative setting, build upon previous knowledge, and relate concepts to real-world scenarios. [17, 18]. Recent studies in both first-year and upper-level engineering courses demonstrated that teaching with an emphasis on active learning allows for more technical content, while at the same time improves student learning and their performance [9].

Examining the situational factors that influence student learning, course learning objectives, the EMP bridge course has been designed to increase the active and collaborative learning activities, integrate engineering concepts from multiple disciplines, increase student-faculty interactions and student performance. Unlike the expensive active-learning, flipped classrooms, this course has been taught in a traditional classroom with nominal technology such as tablet computer to record videos and screencast real-time problem solving, and a learning management system such as Blackboard.

Eligibility requirements for the EMP bridge course were equivalent to that of the university's standard college algebra. In two academic years (2014–15, 2015–16), 233 students have enrolled in this class. While a few of them had been exposed to pre-calculus, the majority had no experience in high-school or college. Although the course was designed for students who had previously experienced college algebra, exceptions were made for students who placed higher but felt more comfortable taking an introductory bridge course before attempting any other engineering courses. The students self-registered themselves with no particular recruitment strategy. This three-credit hour course met twice a week for 1 hour 50 minutes over a 16-week semester, with the same instructor, textbook, syllabus, and similar homework and exams. While this course was designed to bridge the gap between engineering, mathematics, and programming concepts, it was not designed to replace the need for core mathematics courses that are typical in engineering curriculum.

The broad nature of concepts covered in the class required careful planning to promote learning across faculty demonstrations, thing-pair-share activities, brainstorming sessions, and hands-on activities. The first ten minutes of each class was used to reflect on lessons learned from the previous lecture, and briefly discuss current trends and technology advancements. This served as an icebreaker and set the students in learning mode. The next ten minutes were spent on clear misconceptions from previous class session, as they formed a bridge to next course topics, and served towards knowledge retention [19]. During the next hour or so, students engaged in numerous activities such as brain storming sessions, faculty-student interactions, thing-pair-share activities, extrapolating theory to real-world examples, and hands-on design activities. The technical concepts covered ranged from simple composition of a resistor, simple algebraic equation, to second-order RLC circuits, and energy delivered by an integrating operational amplifier as in Table 1. These broad range of activities helped students make connections to what they have learned previously, relate to new concepts, and build a foundation that could retain for a long time. At the end of each class session, each student group is requested to reflect on what they have learned during class and what concept required more explanation. Accordingly, each student group documented three topics they have learned, and one-two topics that required further coaching. The instructor compiled this information, and updated the following lecture to close the loop, and reinforce student learning. This documentation also served as a study guide for midterm and final exams. The first ten weeks of the course were dedicated to fundamental concepts in engineering applications, mathematics, and computer programming, and last five weeks are used for in-depth concepts such as integral calculus, first and second-order RLC circuits, control structures, and optimization functions.

This course provided students with a significant number of electrical engineering and mechanical engineering problems, and extensive opportunities for active learning during the class period. Accordingly, lab activities were designed and introduced to students to provide them with hands-on experience on an engineering application concurrently with the theoretical concepts they learned during the lecture period.

Problem-solving being the fundamental skill engineering students need to develop, leveraging on the PBL methodologies, each new concept has been taught in four steps: (1) Modeling: transferring physical problem into a math equation involving assumptions; (2) Math solution: using proven methods to solve math equations in step-1; (3) Numerical Solution: demonstrate how computer programming could be used to reduce the time to solve a problem; (4) Physical Meaning: discuss the meaning of solution obtained, and updates required for optimiza-

Table 1. Lecture and lab topics in the EMP bridge course

Week	Lecture	Lab
1	• Voltage and Current relationship in a car battery, operational amplifier • Order of precedence in scalars • Elementary math functions using simple computer programs	Introduction to MATLAB
2	• Equivalent resistance, inductance in series and parallel circuit • Creating and executing a computer script file	Executing simple MATLAB programs
3–4	• Kinematics using one-link and two-link planar robots • Impedance in an RL circuit • Variables and their assignment • Vectors and arrays	Relationship between voltage, current, resistance in batteries and wires
5–6	• Voltage phasor diagrams in RL circuits • Midterm exam	Kinematics and Inverse Kinematics through one-link and two-link robots
7	• Impedance of RLC series and parallel circuits • Armature current in a DC Motor • Element-by-element calculations in computer programs • Creating two-dimensional plots	MATLAB hands-on activity to reinforce knowledge in vectors, array, and two-dimensional plotting
8–9	• Angular motion of one-link planar robot • Voltage relationships in an operational amplifier circuit with sinusoidal supply voltage • Relational and logical operators • Conditional Statements	Measurement and analysis of harmonic signals in circuits
10–11	• Two-loop circuits as applicable to household • Programming Loops • Midterm exam	Design, implement and optimize two-loop circuit as applicable to household
11–12	• Impact of various supply voltages and currents on an inductance coil and miniature batteries • Plotting polynomials, performing curve fitting, and interpolation	MATLAB hands-on activities to reinforce knowledge of conditional statements and loops, and plotting polynomials
13–15	• Current, voltage, and energy stored in a miniature battery • Performing differentiation and integration to calculate voltages and currents	MATLAB hands-on activities to reinforce knowledge in interpolation, differentiation, and integration as applied to circuit elements

tion. This allowed for students to gain an exposure to higher level engineering problems in the first-year, understand significance of mathematics concepts to solve engineering problems and identify ways to solve the problems swiftly using programming. For example, to demonstrate the significance of two-loop circuits, the system of equations, matrix manipulation, and loops, the following process was followed. An engineer was tasked to help in a havoc area that lost power. S/he was given two car batteries operating at 12V and tasked to operate all ceiling fans and fluorescent light bulbs for a home in Fig. 1. S/he was also given the following parameters: R_{light}= 10 Ω, R_{fan}= 20 Ω. Operation of a light bulb requires at least 1.5V, and ceiling fan requires at least 3V.

In the first step, students were taught to model the house into a circuit diagram with appropriate items as in Fig. 2, and formulate math equations based on Ohm's law and Kirchhoff's Voltage Law as in Equations (1)–(4). In the second step, students were taught to use substitution method and/or Cramer's rule to find relevant voltage drops across each element (V_{R1} = 3.65 V, V_{R2} = 7.30 V, V_{R3} = 1.04 V, V_{R4} = –5.21 V, V_{R5} = –5.21 V, V_{R6} = –2.60 V) to identify if a solution has been obtained. Comparison with the design constraints was performed, students learned why the current configuration does not work to operate all devices, and discussed potential solutions such as reversing the polarity of battery placements.

In the third step, students were taught how to translate the system of equations into matrices, and write a MATLAB program to solve for voltage drops using matrix inversion method ($x = A^{-1} \times b$). As this process gave the same solution as in the previous step, students updated the program in fourth step to loop the program for four times, with each iteration calculating voltage drops for a different battery polarity. Accordingly, they found a feasible solution with acceptable voltage drops (V_{R1} = 2.60 V, V_{R2} = 5.21 V, V_{R3} = 4.17 V, V_{R4} =3.13 V, V_{R5} = 3.13 V, V_{R6} = 1.56 V), and updated the battery configuration as in Fig. 3. Students further reinforced this concept in the hands-on design activity, where they have built this circuit on a prototyping breadboard, ensured that the measured values are consistent with the calculations. This systematic process of problem solving followed by reinforcing through hands-on activity helped students understand significance and application of two-loop circuits in electrical engineering, arrays and loops in computer programming, and matrix inversion and Cramer's rule in mathematics. Students further explored how the same systematic process could be applied to solve problems in other applications such as signal processing, statics, dynamics, and robotics.

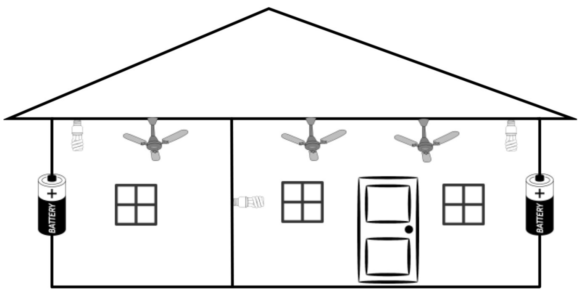

Fig. 1. Example model house used to teach System of Equations and Programming Loops.

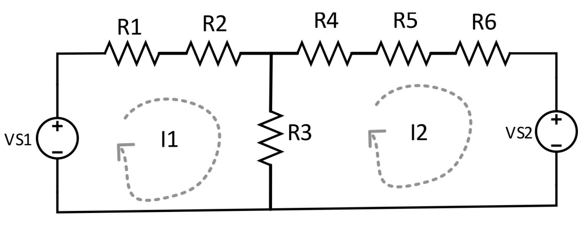

Fig. 2. Two-loop circuit that mimics operation in a model house.

$$VS1 = I1(R1 + R2 + R3) = I2 \times R3 \quad (1)$$

$$VS2 = -I1 \times R3 + I2(R3 + R4 + R5 + R6) \quad (2)$$

$$12 = I1 \times 40 - I2 \times 10 \quad (3)$$

$$12 = -I1 \times 10 + I2 \times 60 \quad (4)$$

Through working on problems that relate to real-world scenarios, students were very much engaged in all activities during the course. They have appeared to gain a sense of pride after completing each activity in the course. After each class session, students took a homework/quiz to evaluate their understanding of the course material. The immediate feedback provided helped students gain confidence, and helped instructor gain a better grasp on what concepts students understood and what concepts needed further clarification, which was revisited in the following class session. Overall, this educational method proved to be efficient in retaining student interest during class time, providing timely and relevant feedback, and reinforcing continued student learning through the semester.

3. Methods and evaluation

3.1 Measures

The effectiveness of proposed EMP bridge course is evaluated through pre- and post-survey of student perceptions on engineering mathematics and programming applications, instructional methodologies, exam grades in this course, and follow-up circuit analysis course. Exams in these course offering are constructed using multiple-choice questions, open-ended problems, and circuit design questions that are aligned with course learning objectives. Consistency in exams was achieved across multiple offerings through grading rubrics and using the same instructor to teach courses.

In addition, formative assessments were also conducted. To identify if the instructional methods were effective and which methods require further refinement, timely student reflections are obtained through the semester. After students have taken the first midterm exam in the fifth week, they were formed into small focus groups are asked to collectively reflect on both strengths and weakness of the course. This feedback session helped the instructor identify instructional methods that have the most and least impact, and update as necessary. The same feedback session was held at end of the semester to identify if there were positive or negative trends in student feedback.

As continual evaluation and assessment of each hands-on design activity is important, students were requested to provide feedback on each activity, and extrapolate theoretical concepts to practical real-world applications. This ensured that students could hone their critical thinking and problem-solving skills, and communicate effectively. Lastly, to identify if the course had a positive influence on the student learning beyond the semester, performance of students in this experimental group has been compared with the general population in the follow-up course of circuit analysis.

3.2 Student perceptions

As students self-registered themselves into the course, to establish comparability of the experimental student population and the general population, aptitude variables such as mean high school GPA and mean composite ACT scores are compared and presented in Table 2. This shows that mean and standard deviations for both student groups are similar in high school GPA, but less achieving in composite ACT scores. Beginning of the semester, students were asked to rate their preparedness for engineering undergraduate program based on their high school mathematics and programming classes, with mean values presented in Table 3. This shows that scores are consistent for both academic years, where students stated that they are moderately prepared in engineering mathematics, but consistent for both academic years, where students stated that they are moderately prepared in engineering mathematics, but underprepared in computer programming, further validating that students feel the need to learn programming and its relation to engineering applications.

In order to encourage collaborative learning and team building, during the first day of class, students

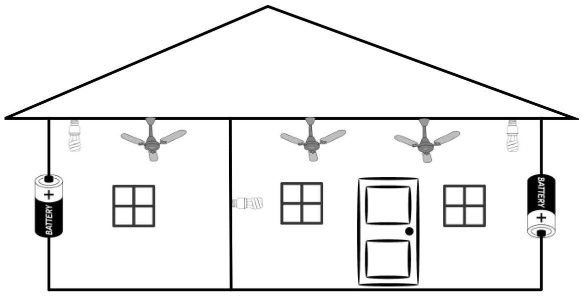

Fig. 3. Model house with updated voltage sources for reliable operation of fans and lights.

Table 2. Comparison of student aptitudes

Group	Experimental	Control	Difference
High-school GPA	3.45	3.60	–0.15
Composite ACT	23.49	25.26	–1.77

Table 3. Student Perceptions on Preparedness

Focus Area	2014–15	2015–16
Mathematics	3.05	2.98
Programming	1.49	1.41

Scale: 1—Low, 5—High.

were engaged in a collaborative learning orientation session and shown the different forms of learning and respective strategies. Right after, the course started where students were exposed to the technical concepts through instructional strategies, hands-on design activities, and a midterm examination. After a third of the semester, during week-06, students were teamed and a formative mid-semester feedback evaluation was conducted to identify working instructional strategies, and refinements necessary to improve student learnings as presented in Table 4. This timely feedback practice [20] proved to be instrumental as it helped the instructor identify where learning is occurring, and what must be done through the remainder of the semester. This positive outcome via a mid-semester feedback is consistent with research findings [21] that frequent consultations with students result in acceptance of instructional practices and improves their learnings.

To improve student understanding of course concepts and engage them in intellectual discussions, several collaborative learning activities are infused throughout the course. These activities provided opportunities to enhance student critical thinking and problem-solving skills vital for success in engineering [22]. At the end of each activity, students were requested to provide brief feedback, with mean and standard deviations for each year (2014–15, 2015–16) summarized in Table 5. Data from both years have provided consistent results, where students have rated the hands-on experience

Table 4. Excerpt of mid-semester student feedback

Strength	Examples
Available outside of class; always available; easy contact with Prof; office hours	He has always been there; easy to reach by email; gave us a survey to determine best practices for everyone
Very deeply! knowledgeable; enthusiastic teaching style	Cares about material; explains concepts well
Examples from future engineering classes	Providing examples from 2nd and 3rd year engineering classes helps us find reasons for learning each concept
Professor is open to questions; clear instructions	Answers any question, no matter how trivial. Provides clear instructions
Many examples; purposely slips up during examples	Do examples wrong for us to catch him and correct him
Dynamic updates to schedule and clarification	Makes sure that we understand before moving, provides real-world scenarios
Notes and videos available before class	Able to get an understanding of topics before class
Handouts; learn by doing; detailed notes	Goes through several examples for each topic; in-class handouts are useful
Real life practical applications; and solving problems using MATLAB	Gives a diverse perspective of how programming can solve problems in different engineering disciplines
Group/partner work	Learning process and the work is made easier with a second mind
In-class demonstrations	Demonstrations helped us confirm that theory connects to practice
Hands-on activities	Labs connected theory to practice and showed how course concepts relate

Table 5. Student feedback on design activities

Question	Year 2014–15	Year 2015–16
How much did you enjoy the labs?	3.69 (1.08)	3.77 (0.93)
How much did you think you learned from the labs?	3.73 (1.08)	3.81 (0.84)
To what extent did the labs require brainstorming?	3.89 (1.10)	3.66 (1.08)
To what extent did the labs require you to use problem-solving skills?	3.65 (1.12)	3.96 (0.91)
Was there sufficient time allocated to complete tasks assigned?	4.40 (1.09)	4.55 (0.75)
How involved do you feel you were in the activities required for this lab?	4.31 (0.87)	4.53 (0.84)
How confident do you feel about your performance on the activities required for this lab?	4.30 (0.80)	4.31 (0.89)
Compared to other group members how much did you contribute to the overall lab exercise?	4.20 (0.94)	4.28 (0.83)

Scale: 1—Low, 5—High.
Note: Cell entities are Mean and standard deviations (in parenthesis).

are enjoyable at an acceptable level. Students have rated their confidence and participation in labs very highly, demonstrating the positive influence these activities had on their learning and academic success.

One of the main objectives of this course is to help students realize the practical applications of mathematics and programming concepts, and how they relate to solving engineering problems. A perception survey was provided to students at beginning and end of the course, with results presented in Fig. 4 and Fig. 5. As most students had a moderate prior knowledge level of mathematics concepts, an acceptable improvement was observed in understanding of how mathematics concepts could be used to solve engineering problems. On the other hand, most students had very low prior knowledge and understanding of programming concepts. After participating in this course, students gained a good understanding of how computer programming concepts could be used to solve practical engineering problems.

3.3 Student performance evaluation

One of the primary objectives of the EMP bridge course is to better educate students on how mathematics and programming could be used to solve engineering problems effectively. Through a systematic process outlined previously and using active learning strategies, student learning has been improved. As this course has been taught several times in two academic years, to identify any statistical variation between the student groups, ANOVA was performed as summarized in Table 6. The statistical distribution of all course offerings showed to follow a normal distribution, with an average GPA of 2.73, and also demonstrated that there is no statistical difference between any of the course offerings.

To further evaluate effectiveness of the proposed course, student performance in a follow-up circuit analysis course was compared for a period of three years. During this time, 233 students have taken the EMP course and obtained a mean GPA of 2.73. Simultaneously, 332 students have also taken the

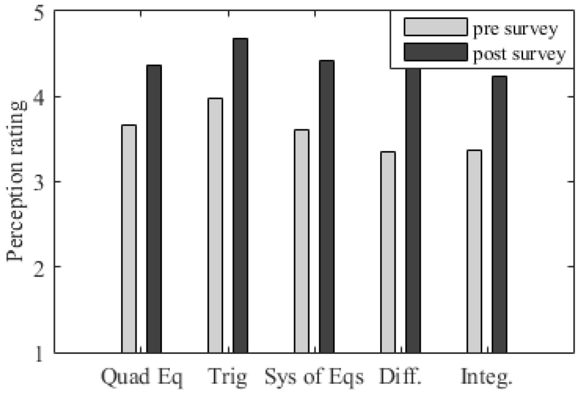

Fig. 4. Change in student perception of engineering mathematics applications.

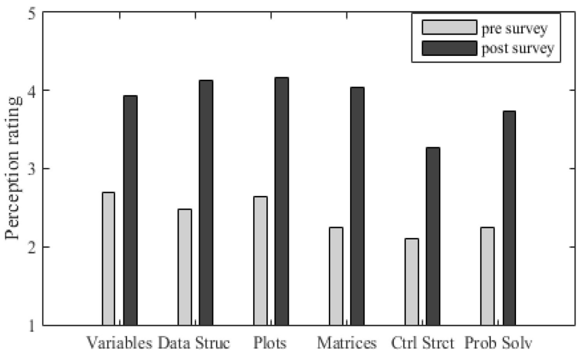

Fig. 5. Change in student perception of programming applications.

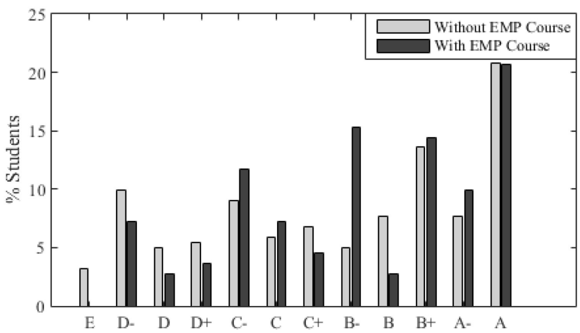

Fig. 6. Student grade distribution in circuit analysis course.

circuit analysis course and obtained a mean GPA of 2.61. Analysis in Table 7 shows that the mean grade of students in circuit analysis has increased by approximately 0.20 because of taking the proposed EMP bridge course. Also, the standard deviation of grades has decreased across all semesters. To evaluate further, distribution of student grades is presented in Fig. 6, and shows two important points. The percentage of students who are at risk of failure (GPA < 2.0) has decreased from 32% to 25%, and percentage of students who needs a motivation and have the potential to succeed (2.0 < GPA < 2.7) have increased from 17.6% to 27%, further demonstrating the promising influence of the EMP bridge course on student academic success.

Table 7. Student performance in circuit analysis course

Group	Average GPA	Median GPA	n
Circuit Analysis without EMP Course	2.54	2.7	221
Circuit Analysis with EMP Course	2.74	2.7	111

4. Discussion

Overall, the proposed first-year bridge course was eventful, encouraging to students, and demonstrated promising results in the current and follow-up courses. These results are particularly worthy, as high attrition and low enrollment have already created additional pressure for engineering educators to better prepare students with longitudinal positive impact on learning.

First goal in the proposed EMP bridge course is to demonstrate the significance of mathematics and programming concepts, and how they could be utilized to solve engineering problems efficiently. This learning goal has facilitated the purposeful alignment of instructional design elements to situational factors including student characteristics, instructor characteristics, influences from external stakeholders, and influence on course context. Based on feedback from students in the pre- and post-survey in Fig. 4 and Fig. 5, students have gained a better understanding of these concepts. Further, student performance in the exams in follow-up courses with an average GPA of 2.74 demonstrates that students gained the problem-solving ability, a much-needed skill in first-year students [2]. This combined with the better performance of these students in comparison with the general population in follow-up class of circuit analysis clearly demonstrates that students can extrapolate their learnings into sophomore courses.

Through the diverse range and number of hands-on activities in, students were able to appreciate the

Table 6. ANOVA test results for student grade distribution

Source of Variation	Sum of Squares	df	Mean Square	F	Sig
Between Groups	0.051	1	0.051	3.569	0.062
Within Groups	1.287	89	0.014		
Total	1.339	90			

practical applications of each concept from various viewpoints and strengthen their content knowledge in fundamental circuit analysis, programming constructs, planar robotics, and were encouraged to pursue a degree in electrical and computer engineering. The combination of providing instructional material such as handouts, informational videos, tutorials prior to class, conducting in-class individual and group problem solving discussions and hands-on activities, and providing periodic and swift feedback on their learnings proved to be very effective and made students to work together, and improved their content knowledge. These findings present a clear alignment of learning objectives, instructional approaches, and evaluation metrics.

Overall, the proposed EMP bridge course has provided several advantages such as effective integration of technology into the student learning platform, improved student-faculty interactions for sustained learning, effective usage of classroom time for individual and group problem solving while representing real-world problems, allowed for more activity based vs. theory based lecture content, allowed for students to learn at their own pace, discuss among each other, and present their findings in a collaborative manner.

5. Conclusions

This study determined the effects of an experimental bridge course for first-year engineering students to foster their knowledge and experience in the application of mathematics and programming concepts to solve electrical engineering problems. From this study we determined that, through a broad range of engineering applications and activities, students gained a better understanding of engineering mathematics and programming concepts, and appeared to have a positive influence on their learning in the current and subsequent courses. Further, this course resulted in improved student awareness of electrical and computer engineering applications, reinforced their critical thinking and problem-solving skills, that are much needed for academic success.

References

1. P. Dębiec, Effective Learner-Centered Approach for Teaching an Introductory Digital Systems Course, *IEEE Transactions on Education*, **66**(1), 2017, pp. 38–45.
2. S. Brown and J. Burnham, Engineering Student's Mathematics Self-Efficacy Development in a Freshmen Engineering Mathematics Course, *International Journal of Engineering Education*, **24**(1), 2012, pp. 113–129.
3. G. Silva-Maceda, P. David Arjona-Villicaña and F. Edgar Castillo-Barrera, More Time or Better Tools? A Large-Scale Retrospective Comparison of Pedagogical Approaches to Teach Programming, *IEEE Transactions on Education*, **59**(4), 2016, pp. 274–281.
4. C. Watson and F. W. B. Li, Failure Rates in Introductory Programming Revisited, *Proceedings of the Conference on Innovation & Technology in Computer Science Education*, June 2014, pp. 39–44.
5. Science and Engineering Indicators 2016, *National Science Foundation*, Available: https://www.nsf.gov/statistics/2016/nsb20161
6. A. Vihavainen, J. Airaksinen and C. Watson, A Systematic Review of Approaches for Teaching Introductory Programming and Their Influence on Success, *Proceedings of the Tenth Annual Conference on International Computing Education Research*, Aug 2014, pp. 19–26.
7. R. Faux, Impact of Preprogramming Course Curriculum on Learning in the First Programming Course, *IEEE Transactions on Education*, **49**(1), 2006, pp. 11–15.
8. S. Davies, J. A. Polack-Wahl and K. Anewalt, A Snapshot of Current Practices in. Teaching the Introductory Programming Sequence, *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*, Mar 2011, pp. 625–630.
9. K. Yelamarthi and E. Drake, A Flipped First-Year Digital Circuits Course for Engineering and Technology Students, *IEEE Transactions on Education*, **58**(3), 2015, pp. 179–186.
10. S. Singer, N. Nielsen and H. Schweingruber, Discipline-Based Education Research: Understanding and Improving Learning in Undergraduate Science and Engineering, *The National Academies Press*, Washington DC, 2012.
11. I. Calvo, I. Cabanes, J. Quesada and O. Barambones, A Multidisciplinary PBL Approach for Teaching Industrial Informatics and Robotics in Engineering, *IEEE Transactions on Education*, **61**(1), 2017, pp. 21–28.
12. R. M. O'Connell, Adapting Team-Based Learning for Application in the Basic Electric Circuit Theory Sequence, *IEEE Transactions on Education*, **58**(2), 2015, pp. 90–97.
13. J. Biggs and C. Tang, Teaching for Quality Learning at University, U.K., Berkshire: SRHE & Open Univ. Press, 2007.
14. M. Dal, Teaching Electric Drives Control Course: Incorporation of Active Learning into the Classroom, *IEEE Transactions on Education*, **56**(4), 2013, pp. 459–469.
15. S. Freeman, et. al, Active Learning Increases Student Performance in Science, Engineering, and Mathematics, *Proceedings of the National Academy of Sciences of the United States of America*, **111**(23), 2014, pp. 8410–8415.
16. G. J. Kim, E. E. Patrick, R. Srivastava and M. E. Law, Perspective on Flipping Circuits I, *IEEE Transactions on Education*, **57**(3), 2014, pp. 188–192.
17. D. W. Johnson, R. T. Johnson and K. A. Smith, Cooperative learning returns to college: What evidence is there that it works?, *Change*, **20**(4), 1998, pp. 26–35.
18. Hart Research Associates, Falling short? College learning and career success, *Selected Findings from Online Surveys of Employers and College Students*, 2015.
19. E. Jensen, *Brain-Based Learning*, Thousand Oaks, CA, USA: Corwin Press, 2008.
20. K. J. Gillespie and D. L. Robertson, A Guide to Faculty Development, 2nd edition, Jossey-Bass, 2010.
21. C. J. Finelli, et. al. Utilizing instructional consultations to enhance the teaching performance of engineering faculty, *Journal of Engineering Education*, **97**(4), 2011, pp. 397–412.
22. C. Koproske, The promise and perils of innovation: Part I, *Education Advisory Board*, 2012.

Kumar Yelamarthi received his PhD and MS degree from Wright State University in 2008 and 2004, and BE from University of Madras, India in 2000. He is currently a Professor of Electrical & Computer Engineering at Central Michigan University. His research interest is in the areas of Wireless Sensor Networks, Internet of Things, autonomous adaptive systems, embedded systems, and engineering education. He has published over 130 articles in archival journals and

conference proceedings in these areas. He has served as a technical reviewer for several IEEE/ASME/ASEE international conferences and journals and as a reviewer for numerous funding proposals. He served as the general chair for 2016 ASEE NCS Conference, 2011 ASEE NCS conference, technical committee member is IEEE ISVLSI, IEEE MWSCAS conferences. He served as PI, co-PI, and senior personnel in several externally funded grants from organizations such as NSF, NASA, and the regional industry. He is an elected member of Tau Beta Pi engineering honor society, and Omicron Delta Kappa national leadership honor society, and a senior member of IEEE.