Guided Problem Solving and Group Programming

A Technology-Enhanced Teaching-Learning Strategy for Engineering Problem Solving

Abhinav Anand
IDP Educational Technology
IIT Bombay
abhinavet@iitb.ac.in

Aditi Kothiyal
IDP Educational Technology
IIT Bombay
aditi.kothiyal@iitb.ac.in

Bipin Rajendran
Electrical Engineering
IIT Bombay
bipin@ee.iitb.ac.in

Sahana Murthy
IDP Educational Technology
IIT Bombay
sahanamurthy@iitb.ac.in

***Abstract*— An important goal of undergraduate engineering education is to develop students' ability to solve complex, real-world problems. Such problems involve the building of physical and mathematical models of real-world scenarios, identifying correct parameters of the model, devising numerical solution methods, and optimizing the solution. In typical engineering courses, solution methods are demonstrated in problem-solving tutorials while programming is relegated to homework or lab, wherein students may not get ample feedback on their choices of optimization or solution evaluation. In this paper, we propose a strategy, Guided Problem Solving and Group Programming (GPGP) to overcome this gap. Students work in peer groups within class, to build and implement their mathematical models and solutions, then write programs to do optimization and evaluation. We implemented this strategy in a 4th year electrical engineering course in which GPGP was done four times over the semester. We assessed students' performance on dimensions of problem solving such as representing the problem, developing solution, making justification for proposed solution and monitoring and evaluating problem space and solutions across the semester and found a statistically significant improvement. Further students perceived that they had learned engineering problem solving via GPGP and reported enjoying the strategy in class.**

Keywords-engineering problem solving; GPGP; PBL; MEA; scaffolding;question prompts;peer interactions

I. INTRODUCTION

With the addition of new occupational specialties globally and the emergence of new interdisciplinary areas, there exists a great demand for well trained engineers. An important aspect of engineering training is the ability to solve problems of the real world [1]. This is also specified by ABET as program learning outcome (3.e), as "an ability to identify, formulate, and solve engineering problems" [2]. Hence, it is important for undergraduate engineering students to learn engineering problem solving skills before graduating [2].

The well structured problems that students typically solve in the classroom are substantially different from the engineering problems they face in the real world [3]. Engineering problems are ill-structured and have characteristics like conflicting goals, multiple solutions, and unanticipated problems, involve significant amount of mathematics and multiple forms of problem representation [3]. Domain knowledge alone is not sufficient to solve these problems [4] and they require several other cognitive and affective abilities like metacognition, epistemic beliefs, justifications skills, etc [5].

In order to develop problem solving skills, students must learn conceptual knowledge and model building of real world scenarios from physical principles and data [6]. Solutions to engineering problems are iterative in nature and several iterations are needed to arrive at an optimal solution [7]. Further, very often it is extremely expensive or even impossible to find the analytical solution to engineering problems and a numerical solution suffices for the situation [8]. Therefore engineering students need to learn to solve engineering problems numerically. This includes translating mathematical and physical models to the language of computation, writing the program to determine a numerical solution and optimizing it [8], [9].

Several instructional strategies have been used to teach engineering students open ended problem solving skills. Problem based learning (PBL) [6], model eliciting activities (MEA) [10] and the McMaster University problem solving program [11] are some such strategies which have been employed in engineering classrooms widely. The McMaster program implemented across multiple courses over four years, has proven extremely successful in developing engineering students' problem solving skill, via repeated implementations and evaluations. However it is not clear what the learning effectiveness of a one semester module would be, were we to implement it in isolation. Owing to its roots in medical education, the problem solving process in PBL is designed to help students develop the hypothetico-deductive problem solving model, i.e., hypothesis generation and evaluation, and learning specific content needed to solve the problem [12]. In MEA, the emphasis is on conceptual understanding via model development rather than problem solving skills [13]. Thus neither of these teaching-learning strategies is directly applicable to the teaching of engineering problem-solving whose characteristics we have defined above.

In this paper, we propose an intervention named as Guided Problem solving and Group Programming (GPGP), designed to develop engineering students' problem solving skill. This strategy borrows the conceptual framework for ill-structured problem solving instruction presented in [14] and derives several elements from PBL and MEA. It rests on the dual scaffolds of question prompts and peer interaction recommended in [14]. Thus the entire problem solving process is performed in class in a scaffolded manner so that students became aware of these processes. The contribution and strength of our intervention lies in the fact that the most critical aspect of engineering problem solving namely writing the code for the numerical solution, testing and optimization it, which is usually performed by students in homework or labs, is also performed in class. By engineering problems we mean open ended, ill-structured and real world based problems. GPGP has been implemented in a field setting in a course which aims to prepare students to do their senior projects. Students specifically learn engineering problem solving via several numerical methods. They solve engineering problems in the classroom and try to arrive at an optimal solution by implementing numerical methods in a programming environment of their choice.

In this study, we evaluated our intervention in a field setting to answer the following two research questions:

RQ1: Does GPGP strategy improve engineering students' problem solving skills?

RQ2: What are students' perceptions of learning problem-solving via GPGP strategy?

Since this is the first time we are evaluating this strategy we wished to understand students' perceptions of learning with GPGP which motivated RQ2. We performed a pre-experimental, single group pre-post test design to answer RQ1 and conducted a questionnaire to answer RQ2. We found that student performance on several dimensions of problem solving skill, namely, defining the problem, identifying relevant information, selecting or developing solution, quality of solution and constructing argument, improved significantly. Further 73.33% students agreed that they learned engineering problem solving via GPGP strategy and 93.33% students reported enjoying the class. Instructor perceptions confirm that GPGP was beneficial to students to learn engineering problem solving.

Our contribution in this work is an intervention strategy to teach engineering problem solving that incorporates many features known from literature to improve ill-structured problem solving ability. In addition, by incorporating in-class group programming in our strategy, the crucial step of engineering solution implementation and evaluation via numerical methods is brought into the classroom, from its usual place in homework or lab, thus strengthening our instructional strategy.

II. BACKGROUND AND RELATED WORK

Problem solving is one of the fundamental learning goals of engineering education [2], [3]. There are two kinds of problems that are encountered in engineering domains, ill-structured and well-structured. A student frequently faces well-structured problems in traditional classroom in contrast to the ill-structured or open ended problems which are more prevalent in everyday and professional practice [3]. Students find themselves under prepared for workplace open ended problem solving [3] and their inability is because of failure to apply knowledge from one context to another [14]. The broad goals of our course are to train students to do their senior projects. The focus is to develop students' problem solving skill, specifically in learning how and when to use a set of numerical methods and mathematical tools to solve real world problems. Below we review interventions available for these goals and explain the need for our new strategy.

Over a period of two and a half decades, Don Woods and his colleagues at the McMaster University in Canada [11] have defined problem solving skill, identified a set of effective teaching methods to develop this skill, implemented it as a series of four required courses and evaluated its effectiveness. This teaching approach focuses on developing each dimension of the problem solving skill separately. As part of their problem solving program, students take workshops to develop these skills beginning from the second year. The first course teaches well-defined, back of the textbook problem solving skill. The next course focuses on giving additional practice in the application of these skills. The third course teaches group problem solving while the final course in the final year is targeted towards developing the skills needed for open-ended problem solving and lifelong learning. The authors [15] found that merely providing the students with problem solving practice or working out problems in class is ineffective in improving problem solving skills. Hence they developed a set of activities based on research in cognitive and behavioral psychology, with each activity focused on one particular skill. While this is a highly effective program for training effective engineering problem solvers, its scope is too wide for our goals.

Problem based learning (PBL) is a well-established pedagogy which helps students in developing deep conceptual understanding and learning real world problem solving skills [12], [16], [17]. It is an active learning, student centered strategy developed to prepare medical students for problem solving in clinical settings [16]. PBL is a pedagogy based on constructivism and includes inquiry activities, collaborative learning and self-directed learning [12], [17]. It has been shown that PBL implemented in an electrical engineering course resulted in higher learning gains compared with traditional lecture [6].

Model eliciting activities (MEA) aim to improve conceptual understanding and problem solving skills by encouraging students to model real world situations and analyze these models. It is an instructional strategy that presents students with a thought revealing, open ended and inquiry driven problem [18] that students solve in groups. Originally, developed for mathematics education, MEAs have become popular in engineering education for a range of learning outcomes. Engineering education research has shown that using MEAs lead to improvement in conceptual understanding and problem solving ability [15], [19] and serves as a method for assessing student problem solving process [20].

Numerical methods are becoming increasingly important in science and engineering as these disciplines involve the interaction between theory, experiment and numerical methods [8], [24], and the growth of computational power has made the implementation of numerical methods cheaper. In fact, many engineering problems cannot be solved except by numerical methods [8]. Such methods are a way for engineers to convert the mathematical and physical representation of the problem into information that they can use to make engineering decisions [8]. Despite their importance, however, the teaching of the programming aspect of numerical methods is relegated to homework or lab in courses, be they numerical methods courses (for eg. see [21], [22]) or design courses (for eg. see [7], [9]), and students mostly work independently to write code.

Jonassen [4] has proposed that learners be guided through all the steps of the open-ended problem solving process using appropriate scaffolds at each step in order to improve their overall problem solving skill. The role of instructor guidance and regular feedback as factors affecting the success of MEAs was found in [13]. Similarly, even in PBL the facilitator scaffolds student learning through the use of meta-cognitive question prompts [12]. Therefore, it follows that in order for students to develop the skill of using numerical methods to solve engineering problems, they must be scaffolded in this process as well.

In this paper, we have developed an intervention based on Jonassens' ill-structured problem solving instructional design [4] which is adapted in the theoretical framework of Ge [14], [23], [26]. This framework recommends two kinds of scaffolding strategies implemented differently at different stages of problem solving, namely question prompts and peer cooperation [6]. Like PBL and MEA, students are presented with an engineering problem that they solve by creating a mathematical model and then use appropriate numerical methods to develop and optimize their solution. Our intervention differs from previous interventions in the fact that the two scaffolds extend into the programming part of the problem solving process and students are also guided as they write the code, test and optimize it. The details of the intervention are presented in the next section.

III. GPGP STRATEGY.

The instructor used a technology enhanced teaching learning strategy to teach while students learned how to solve engineering problems using various numerical methods. Students implemented and optimized their solutions in software during in-class activities. The intervention strategy is termed Guided Problem solving and Group Programming (GPGP) and has two scaffolds for students as they attempt to solve the problems [14], [23], namely, (i) instructor question prompts to the entire class and (ii) peer interactions which occur in the student groups.

The class began with the instructor introducing a semi-structured problem. Once this problem had been solved he increased the complexity towards the engineering problem. Students solved the problem in pre-assigned groups. The whole intervention process can be broadly divided into three phases after the problem is presented to the students. The two elements of scaffolding remain prevalent across the three phases as described below. The overall flow of the strategy in a typical class can be represented by Fig. 1.

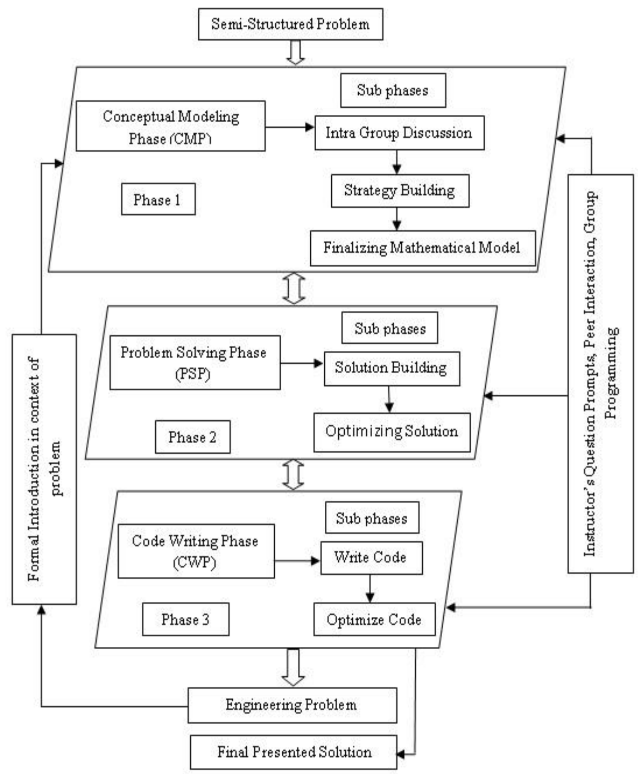

Figure 1 GPGP flow chart

A. *Conceptual modeling phase (CMP)*

This is the very first phase of the intervention strategy. Intra group discussion, strategy building and finalizing mathematical models are the sub-phases of this phase. By the end of CMP the group finalizes a mathematical model which it will apply to solve the semi-structured/engineering problem. From this phase students are expected to develop the skills of representing the problem (model building) and interpreting and analyzing the data (identifying relevant information). This phase of the strategy is constantly supported by instructors question prompts and peer interaction.

B. *Problem solving phase (PSP)*

This is the second phase of the intervention strategy where the student groups work towards applying their mathematical models to build a solution for the semi-structured/engineering problem. Solution building and optimization are the two sub phases of this PSP. Students are expected to learn the skills of developing a solution (choosing the appropriate numerical method) during this phase. This phase also has scaffolding in the form of question prompts from instructor and peer interaction.

C. *Code Writing Phase (CWP)*

This is the last phase of GPGP strategy where students implement their solutions in a programming environment and vary the underlying variables and parameters to optimize their solution. Writing and optimizing code are the two sub-phases. Students are expected to learn the skills of writing code for the numerical method (developing the solution), evaluating solutions (testing code) and assessing alternate solutions (optimizing code) during CWP. Since all students have laptops, they write the code in the classroom and the scaffolds of instructor questions and peer interactions remain. Thus using laptops this strategy brings the lab into the classroom. Further, open discussion of all solutions developed in the class happens during this programming phase which offers important feedback to the groups regarding the quality of their solutions.

Students were allowed to revise their preliminary models and programs using the feedback from the instructor and their peers. Thus the student problem solving was iterative and multiple iterations were needed to arrive at a working solution.

GPGP can be applied by engineering instructors to teach students problem solving where programming is an important part of the solution development. Writing programs will help student arrive at an optimal solution and assess the best solution among multiple solutions.

IV. IMPLEMENTATION

The participants of the study were 28 students enrolled in "Foundations of Projects" course which is a course for fourth year electrical engineering undergraduates. The class meets once a week for 90 minutes and the course ran for 10 weeks. The learning objectives of the course were:

1) Develop intuition about model building from physical principles and data
2) Develop hands-on experience to solve various numerical methods problems
3) Understand how to interpret and analyze data
4) Write programs to analyze situations involving random processes.
5) Write programs to solve differential equations numerically.
6) Write programs to solve systems of equations numerically.
7) Write programs to analyze and interpret common distributions.
8) Write programs to determine least squares estimates for data

The programming platform was of student's choice namely, MATLAB, Python, C++ etc. The lectures were taken by the instructor and he gave emphasis on the above mentioned learning objectives. The timeline for the various phases of activity of GPGP strategy that instructor has followed are:

1) Conceptual Modelling phase (CMP): 15-30 minutes
2) Problem solving phase (PSP): 30-40 minutes
3) Code writing phase (CWP): 15-20 minutes

The instructor began the activity by posing simple question prompts in the class and increased the complexity of question prompts as the class progressed. Students first solved the semi-structured problem in groups of three students and followed all three phases of GPGP strategy for every problem. Finally, the instructor posed an engineering problem building upon the complexity of previous given problems. The problems were from common distribution, least square estimates, random motion etc. some of the Typical example of a semi-structured problem posed in the class by the instructor is:

"Q. 1 Create a set of N = 1000 uniformly distributed random numbers in the interval [0; 1]. How do you know the numbers are uniformly distributed?
Q. 1.1 Repeat your exercise for N = 10; 100 and 10000.
Q. 1.2 Create a set of 10000 random numbers in the interval [0; 1], such that the probability of occurrence of the value x is linearly proportional to x."

Examples of engineering problems are:

"Q.2 What is the sample size required to reliably predict an election?
Gotham city has 10^6 residents. In an election for the post of Commissioner, Mr. Gordon won 55% of the votes. Write a MATLAB code to create a voting pattern such that Mr. Gordon won 55% of the votes.
Q. 2.1 Now, randomly chose a sample of s = 100 people from the residents. If you had chosen this particular sample to conduct an opinion poll before the election, what would your prediction be regarding Mr. Gordon's chances of winning the election?
Q. 2.2 Create 1000 such samples, each sample containing s = 100 randomly chosen people, and create a raster plot for the probability of Mr. Gordon's chances.
Q. 2.3 Plot the histogram of the 1000 sample probabilities and determine its mean and variance.
Q. 2.4 Repeat the experiment with s = 500; 1000; 2000; 3000."

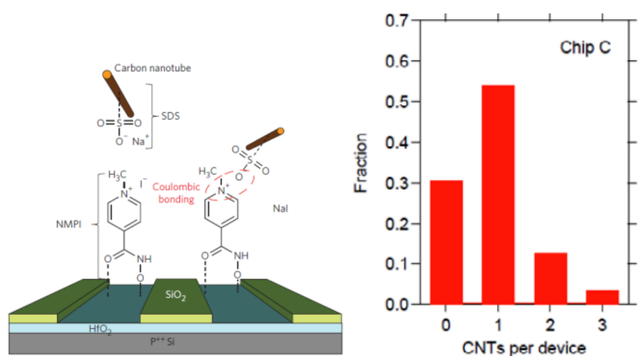

Figure 2. Engineering problem on common distribution function

"Q. 3 Researchers at IBM claim that they have a magic recipe to localize the growth of CNT in trenches. The statistics of CNTs per trench obtained on a chip is shown on the right. Did the experiment succeed? Assume there are 256 tubes and 256 trenches in the experiment." (Refer Figure 2)

All the problems are scaffolded by instructor through question prompts. The instructor gave question prompts at five places during the problem solving process:

1) During conceptual modeling phase (CMP)
   i) Question prompts while discussing problem representation.
2) During Problem solving phase (PSP)
   ii) Question prompts while discussing solution development.
   iii) Question prompts while making justification for proposed solution
3) During code writing phase (CWP)
   iv) Question prompts while optimizing the solution
   v) Question prompts while assessing alternate solutions.

Typical examples of instructor question prompts are:

1) How real data looks like?
2) Can we come up with a mathematical model for this physical data?
3) How do you create exponentially distributed set of random numbers?
4) Think about it physically? What is the probability that you are going to go very far without changing directions?
5) Is there anything that you can do to make convergence faster?

V. RESEARCH METHOD

A. Sample

The participants of the study were 15 out of 28 fourth year undergraduate students enrolled in "Foundation of Projects" course offered by the Electrical Engineering department at our institute. These students were admitted to electrical engineering major after they qualify extremely competitive exam testing analytical skills in Physics, chemistry and mathematics and were among highest rankers. Hence, all the students in the study can be considered as equivalent in all respect, except prior exposure to different programming platforms.

B. Instruments

To answer RQ1, "Does GPGP strategy improve engineering students' problem solving skills?", pre- and post-tests, each consisting of a single open-ended engineering problem, were evaluated using a rubric validated in [23]. The rubric measures students' open-ended problem solving process across various dimensions such as: a) Representing the problem b) developing solution c) making justification for proposed solution and d) monitoring and evaluating problem space and solutions. The rubric has sub-dimensions that measures specific problem solving skills and is shown in Table I below. In this rubric the items of "Selecting or developing solution", "Evaluating solution(s)" and "Assessing alternate solution" are related to writing code, testing and optimizing it. Each sub-dimension is scored on an ordinal scale with levels 0-1-2-3 or 0-2-4 as recommended in [23].

TABLE I. SCORING RUBRIC

Scoring rubric for engineering problem solving	
Skills	***Sub-Skills***
Representing the problem	a) Define the problem b) Generate sub goals c)Identify relevant information d) Seek needed information
Developing solution	a) Selecting or developing solution b) Quality of solution
Making justification for proposed solution	a) Constructing argument b) Providing evidence
Monitoring and evaluating problem space and solutions	a) Evaluating solution(s) b) Assessing alternate solution(s)

We administered a questionnaire to all the students to answer RQ2, "What are students' perceptions of learning problem-solving via GPGP strategy?" The instrument had questions on the following constructs, students' perception of usefulness of the strategy for learning, students' enjoyment of the strategy, students' perceived benefits and students' perceived drawbacks. These constructs were chosen because together they highlight the various aspects of students' perceptions regarding GPGP strategy. Five questions on 5-point Likert scale (Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree) and four open ended questions were asked. The Likert scale questions from the questionnaire were:

1) I learned to solve engineering problems because of GPGP strategy used in the course.
2) The question prompts helped me learn to solve engineering problems.
3) I will be able to solve engineering problem such as the one shown above because I learned via GPGP strategy in EE590 class.
4) I enjoyed the GPGP strategy in EE590 class.
5) The concepts I learned in mathematics class like MA205 are enough to solve engineering problems.

The open ended questions from the questionnaire were:

1) Discussing with my classmates during problem solving helped me learn to solve engineering problems because………..
2) Writing program for my strategy helped me arrive at an optimal solution to the engineering problem because………..
3) What, if any, are the other benefits of GPGP strategy?
4) What, if any, in your opinion are the drawbacks of GPGP strategy?

C. Procedure of data collection

The pre-test and post-test consisted of an open ended engineering question to assess students problem solving skills across various dimensions mentioned in Table I. Pre test was conducted at the beginning of the course and post test was conducted at the end of the semester. The questionnaire on students' perception was administered online via Moodle at the end of the semester.

D. *Data analysis technique*

The pre-test and post-test scores were ordinal data and repeated measurements on a single sample, hence we used Wilcoxon Signed-rank test to compare the means and to test for statistical significance. For the Likert questions from the questionnaire were determined the frequency distribution and performed content analysis for the open ended questions. The questionnaire results are used for triangulating with the test scores.

VI. RESULTS

A. *Problem Solving Skills*

Out of 28 enrolled students, only 15 students appeared for both pre and post tests. Hence we are only reporting the data of those 15 students in this study. The mean scores of the students in the pre and post tests and the results of the Wilcoxon signed-rank test are shown in Table II.

TABLE II. ANALYSIS OF PRE-POST TEST SCORES

Skills	**Pre Score**		**Post Score**		**p-value**	**Diff. Sig. at p=0.05**
	Mean	***Std. Dev.***	***Mean***	***Std. Dev.***		
1. Representing the problem	2.7	1.5	4.5	1.1	0.004	Yes
a) Define the problem	1.3	0.9	2	0	0.015	Yes
b) Generate sub goals	0.7	0.5	1.1	0.6	0.083	No
c)Identify relevant information	0.8	0.4	1.5	0.6	0.008	Yes
d) Seek needed information	0	0	0	0	1	No
2. Developing solution	2.3	1.2	3.9	1.4	0.011	Yes
a) Selecting or developing solution	1.5	0.7	2.1	0.7	0.039	Yes
b) Quality of solution	0.8	0.6	1.7	0.8	0.01	Yes
3. Making justification for proposed solution	2.8	1.7	3.5	1.4	0.119	No
a) Constructing argument	1.87	0.92	2.4	1.1	0.046	Yes
b) Providing evidence	0.9	1	1.1	0.5	0.527	No
4. Monitoring and evaluating problem space and solutions	0.3	0.7	0.1	0.4	0.257	No
a) Evaluating solution(s)	0.3	0.7	0.1	0.4	0.257	No
b) Assessing alternate solution(s)	0	0	0	0	1	No

We find that there was a significant improvement in the students' scores from the pre test to the post test across the dimensions of representing the problem and developing solutions. Zooming into the sub-dimensions, we observe that there was significant improvement in several dimensions such as, defining the problem, identifying relevant information, selecting or developing solution, quality of solution and constructing argument.

B. *Student Perceptions*

The sample for the questionnaire was the same 15 students who answered both the pre and post tests. The summary of student responses is in Table III. In this table we sum up the responses that were strongly agree and agree and those that were strongly disagree and disagree.

TABLE III. LIKERT SCALE QUESTIONNAIRE QUESTION RESPONSES

Questionnaire questions	**Agree% (No. of students)**	**Neutral% (No. of students)**	**Disagree% (No. of students)**
Q.1	60 (9)	33.33 (5)	6.67 (1)
Q.2	86.67 (13)	13.33 (2)	0 (0)
Q.3	73.33 (11)	26.67 (4)	0 (0)
Q.4	93.33 (14)	6.67 (1)	0 (0)
Q.5	26.67 (4)	40 (6)	33.33 (5)

The results show that 60% of the students stated that they learned engineering problem solving due to the GPGP strategy used in the course (Q.1). However when given a specific problem and asked if they would be able to solve it because they learned via the GPGP strategy (Q.3) that number increased to 73.33%. Further 86.67% of the students agreed that the instructor question prompts were helpful in learning how to solve engineering problem solving (Q.2) and 93.33% agreed that they enjoyed learning with the GPGP strategy (Q.4). Finally when asked to judge whether the fundamental mathematics concepts were sufficient to solve the specific given problem (Q.5) students were mostly (40%) neutral.

Content analysis of the open ended questions along the constructs of problem-solving skills, social skills and enjoyment yielded the results shown in Table IV.

VII. INSTRUCTOR PERCEPTION

Having spent 6 years in the industry, the instructor (the 3rd author of this paper) had noticed that many graduates who join the work-force after obtaining an undergraduate degree in engineering, lack engineering skills to solve real-world problems. These include ability to develop quantitative models of physical situations, come up with hypothesis, and verify those quickly using computational tools. The GPGP strategy was developed by the authors of this study to address these problems in a preparatory class for senior undergraduate students in Electrical Engineering to write a thesis in their chosen field in the broad domain of Microelectronics. The instructor spent about 3-6 hours preparing for each 90 minute lecture as it involved developing new problems with increasing complexity and relevance to the skills being taught in class.

TABLE IV. CONTENT ANALYSIS OF OPEN ENDED QUESTION ON STUDENTS' PERCEPTION

Question	Benefits (Frequency)
Discussing with my classmates during problem solving helped me learn to solve engineering problems because:	1. Learning multiple solutions to the problem (11) 2. Learning new ideas. (5) 3. Discussion helps in reaching conclusions. (3)
Writing the program for my strategy helped me arrive at an optimal solution to the engineering problem because:	1. Solution optimization (5) 2. Solution evaluation, parameter variation and constructing deeper understanding of the problem (4 each)
What, if any, are the other benefits (not mentioned above) of the GPGP strategy?	1. Thinking skills improved such as problem solving, modelling real world problems, etc (8) 2. Social skills improved such as listening, participating in discussions (3) 3. Enjoyment of class (3)
What, if any, in your opinion are the drawbacks of GPGP strategy?	Issues due to adverse group dynamics like one person dominating, students falling behind, everyone not participating etc. (7)

The instructor felt that the students responded with a higher degree of interest and alacrity to the problems, prompts and general discussion in class. It was also observed that the discussions within the group enabled students weaker in programming/analytic skills to participate in the learning process more actively.

The instructor also felt that in-class programming strategy adapted in this study is ideally suited for most subjects that involve problem solving using computational methods. It was also felt that this enables students to actively participate in the learning process, and the opportunity to discuss/learn from peers in the class can go a long way in improving the overall class performance. Further, the GPGP strategy also enabled the instructor to quickly identify the outliers in the student group, and could give more attention to weaker students during the interaction process.

The instructor hence felt that though a significant effort has to be spent in preparing the instructional materials for the class, this strategy is highly beneficial for the teaching-learning process and can enrich it significantly for both the instructor and students, especially since all students today bring their laptops or other handheld computational devices to classes.

VIII. DISCUSSION

Our first research question ": Does GPGP strategy improve engineering students' problem solving skills?" was answered by the result of our single group pre-post study which established that students performance improved statistically significantly on the dimensions of defining the problem, identifying the relevant information, developing solution and in constructing argument to their presented solution. We also observe that student scores on other dimensions such as generate sub-goals, seek needed information & providing evidence showed improvement, though the difference was not statistically significant.

Our second RQ "What are students' perception of learning problem-solving via GPGP strategy?" showed that 73.33% of students agreed that GPGP strategy helped them learn problem-solving and 93.33% enjoyed learning with this strategy. Further content analysis of the open-ended questions showed that programming helped students in optimizing the solution, evaluating their strategy, varying the parameters to reach an optimal solution and construct deeper understanding of the problem. Other perceived student benefits were development of thinking skills, social skills and enjoyment. The major drawbacks of the strategy perceived by the students are due to adverse group dynamics like non-uniform participation. Thus the student perceptions triangulate well with the test scores. The results of our study confirm the findings of previous work [14], [17] regarding the roles of instructor prompts and peer interaction in all steps of the problem solving process.

Since this study was performed in a field setting, in the interest of student fairness, we didn't perform a controlled experimental study and instead chose to evaluate our students via a single group study. Thus the potential threats to validity like maturation – in this case actually working on their projects simultaneously –remain. Another threat is the effect of prior problem solving ability of the participants. In order to assess the role of these factors we propose to carry out experimental studies in the future. Another key limitation of the study is that the sample size is small. This limitation could have implications such as: an outlier causing large impact on the result and weaker generalizability.

To address the above limitations our study included multiple data sources for triangulation. Results from students' perception questionnaire and instructor perception supported the pre-post improvement in learning outcome, indicating that GPGP strategy is effective. Hence, despite these limitations, we recommend that this strategy can be applied in engineering classrooms for teaching problem solving. With laptops becoming more and more prevalent, there is no need to separate lab and lecture if programming is required as part of the solution. Students are able to do it right there in the classroom and get a feel for not only the theoretical aspects of a solution but also how it behaves under various numerical conditions. Based on our results, we recommend that instructors can integrate the programming aspect of doing engineering into the classroom in addition to assigning it to lab and homework. The in-class problem solving and programming ensures that students get to practice an important engineering skill and get feedback from their peers and the instructor.

REFERENCES

[1] E. A. Walker, J. M. Pettit and G. A. Hawkins, "Goals of Engineering Education," American Society for Engineering Education, 1968, pp. 373–377.

[2] ABET Engineering Accreditation Commission. "Criteria for Accrediting Programs in Engineering ," 2010, ABET, Inc.: Baltimore, MD, Retrieved March 22, 2010 from http://www.abet.org/forms.shtml#For_Engineering_Programs_Only

[3] David Jonassen et.al, "Everyday Problem Solving in Engineering: Lessons for Engineering Educators," Journal of Engineering Education, 2006, pp. 139-151.

[4] David H. Jonassen, "Instructional design models for well-structured and ill-structured problem-solving learning outcomes," ETR&D, Vol. 45, No. 1, 1997, pp. 65-94.

[5] V. Shekoyan, "Using multiple-possibility physics problems in introductory physics courses (Doctoral dissertation," Rutgers University-Graduate School-New Brunswick, 2009.

[6] Aman Yadav, Dipendra Subedi, Mary A. Lundeberg and Charles F. Bunting, "Problem-based learning: Influence on student's Learning in an Electrical Engineering course," Journal of Engineering Education, April 2011, Vol. 100, No. 2, pp. 253-280.

[7] C.A. Whitfield, D.W. West, Lowell T. P.E. and John A. Merrill, "A First-year Design Project Software Tool to Emphasize Problem Solving with Computer Programming in the Design Process," ASEE Annual Conference and Exposition, June 2012, pp. 13.

[8] S. C. Chapra & R. P. Canale, "Numerical methods for engineers," New York: McGraw-Hill Higher Education, 2010.

[9] Richard J. Freuler, Audeen W. Fentiman, John T. Demel, Robert J. Gustafson and John A. Merrill, "Developing and Implementing Hands-on Laboratory Exercises and Design Projects for First Year Engineering Students," American Society for Engineering Education Annual Conference & Exposition, June 2001, pp. 3545-3556.

[10] Heidi A. Diefes-Dux, Tamara Moore, Judith Zawojewski, P.K. Imbrie and Deborah Follman, "A Framework for Posing Open-Ended Engineering Problems: Model-Eliciting Activities," ASEE/IEEE Frontier in Education Conference, October 2004, pp. 3-8.

[11] D. R. Woods et. al. "Developing problem solving skills: The McMaster problem solving program," Journal of Engineering Education, 1997, 86(2), pp. 75-91.

[12] J. R. Savery & T. M. Duffy, "Problem based learning: An instructional model and its constructivist framework," Educational technology, 1995, 35(5), pp. 31-38.

[13] Hamilton, E., Lesh, R., Lester, F. R. A. N. K., & Brilleslyper, M. (2008). Model-eliciting activities (MEAs) as a bridge between engineering education research and mathematics education research. Advances in Engineering Education, 1(2), 1-25.

[14] Xun Ge, Susan M. Land, "A Conceptual Framework for Scaffolding Ill-Structured Problem-Solving Processes Using Question Prompts and Peer Interactions," ETR&D, 2004, Vol. 52, No. 2, pp. 5–22.

[15] Renee M. Clark, Larry J. Shuman, Mary Besterfield-Sacre, Tuba Pinar Yildirim, "Use of Model Eliciting Activities to Improve Problem Solving by Industrial Engineering Students," Proceedings of the 2008 Industrial Engineering Research Conference, 2008, pp. 110-115.

[16] Alan J. Neville, "Problem based learning and medical education forty years on," A review of its effect on knowledge and clinical performance, Medical Principles and Practice, 2009, pp. 1-9.

[17] C. E. Hmelo-Silver & H. S. Barrows, "Goals and strategies of a problem-based learning facilitator," Interdisciplinary Journal of Problem-based Learning,1(1), 4.

[18] E. Hamilton, R. Lesh, F. R. A. N. K. Lester, & M. Brilleslyper, "Model-eliciting activities (MEAs) as a bridge between engineering education research and mathematics education research," Advances in Engineering Education, 2008, 1(2), 1-25.

[19] Karen M. Bursic, Larry J. Shuman, Mary Besterfield-Sacre, "Improving student attainment of ABET outcomes using Model-Eliciting Activities(MEAs)," ASEE Annual Conference and Exposition, Conference Proceedings, 2011, pp. 1-13.

[20] Tuba Pinar Yildirim, Larry Shuman and Mary Besterfield-Sacre, "Model-Eliciting Activities Assessing Engineering Student Problem Solving and Skill Integration Processes," International Journal of Engineering Education, Volume 26, Issue 4, 2010, pp. 831-845.

[21] S. R. H. Hoole, S. Sivasuthan, V. U. Karthik, & P. R. P Hoole, "Flip-teaching engineering optimization, electromagnetic product design, and nondestructive evaluation in a semester's course," Computer Applications in Engineering Education, 2014.

[22] http://www2.mae.ufl.edu/haftka/numerical/

[23] Xun Ge, "Scaffolding students problem solving processes on ann ill-structured task using question prompts and peer interactions," A thesis in instructional systems, 2001, pp. 164-166.

[24] R. W. Chabay & B. A. Sherwood, "Matter and interactions," John Wiley & Sons, 2011.

[25] Xun Ge, Susan M. Land, "Scaffolding Students' Problem-Solving Processes in an Ill-Structured Task Using Question Prompts and Peer Interactions," ETR&D, 2003,Vol. 51, No. 1, pp. 21–38.

[26] Brett A. Bixler, "The effects of scaffolding student's problem-solving process via question prompts on problem solving and intrinsic motivation in an online learning environment," Doctoral dissertation, The Pennsylvania State University, 2007.