Function Approximation through an Efficient Neural Networks Method

Dr. Chaomin Luo, Mississippi State University

Dr. Chaomin Luo received his Ph.D. in Department of Electrical and Computer Engineering at University of Waterloo, in 2008, his M.Sc. in Engineering Systems and Computing at University of Guelph, Canada, and his B.Eng. in Electrical Engineering from Southeast University, Nanjing, China. He is currently Associate Professor in the Department of Electrical and Computer Engineering, at the Mississippi State University (MSU). He was panelist in the Department of Defense, USA, 2015-2016, 2016-2017 NDSEG Fellowship program and panelist in 2017 NSF GRFP Panelist program. He was the General Co-Chair of 2015 IEEE International Workshop on Computational Intelligence in Smart Technologies, and Journal Special Issues Chair, IEEE 2016 International Conference on Smart Technologies, Cleveland, OH. Currently, he is Associate Editor of International Journal of Robotics and Automation, and International Journal of Swarm Intelligence Research. He was the Publicity Chair in 2011 IEEE International Conference on Automation and Logistics. He was on the Conference Committee in 2012 International Conference on Information and Automation and International Symposium on Biomedical Engineering and Publicity Chair in 2012 IEEE International Conference on Automation and Logistics. He was a Chair of IEEE SEM - Computational Intelligence Chapter; a Vice Chair of IEEE SEM- Robotics and Automation and Chair of Education Committee of IEEE SEM. He has extensively published in reputed journal and conference proceedings, such as IEEE Transactions on Neural Networks, IEEE Transactions on SMC, IEEE-ICRA, and IEEE-IROS, etc. His research interests include engineering education, computational intelligence, intelligent systems and control, robotics and autonomous systems, and applied artificial intelligence and machine learning for autonomous systems. He received the Best Paper Award in the IEEE International Conference on Information and Automation (IEEE ICIA2017). He is an ASEE, INFORMS, and IEEE member. He is currently an Associate Editor of The 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE-IROS 2019) .

Prof. Zhuming Bi P.E., Purdue University Fort Wayne

Zhuming Bi is a Professor of Mechanical Engineering at the Department of Civil and Mechanical Engineering, Purdue University Fort Wayne, USA. He served as a Senior Engineer at National Institute of Standards and Technology (NIST) of USA (2016), a Senior Project Engineer at Northern Ireland Technology Centre, Queen's University Belfast of UK (2007 – 2009), a Research Scientist at the Integrated Manufacturing Technologies Institute of National Research Council Canada (2003 – 2007), a NSERC Postdoctoral Fellow at Simon Fraser University, Burnaby, BC, Canada (2002 – 2003), and a Visiting Scholar to Nanyang Technological University (2001) and City University of Hong Kong (1997-1998). He was as an Associate Professor at the Department of Manufacturing Engineering, Nanjing University of Science and Technology in China (1996 -1999). He received a Ph.D. degree in Design and Manufacturing from the University of Saskatchewan in Saskatoon of Canada (2002), and a Ph.D. degree in Mechatronic Control and Automation from Harbin Institute of Technology in China (1994). Dr. Bi's research interests are Robotics and Automation, Internet of Things (IoT), Enterprise Systems, and Sustainable Manufacturing. He has published 110 international journal articles with 1749 times of citations by others in Web of Science in these research fields. He was the only awardee for the 'Outstanding Faculty in Research' and 'Featured Faculty in Research Endeavors' at Purdue University Fort Wayne in 2016-2017. He received IEEE Region 4 Outstanding Professional Award in 2018.

Dr. Wenbing Zhao, Cleveland State University

Dr. Zhao is a Full Professor at the Department of Electrical Engineering and Computer Science, Cleveland State University (CSU). He earned his Ph.D. at University of California, Santa Barbara in 2002. Dr. Zhao has a Bachelor of Science degree in Physics in 1990, and a Master of Science degree in Physics in 1993, both at Peking University, Beijing, China. Dr. Zhao also received a Master of Science degree in Electrical and Computer Engineering in 1998 at University of California, Santa Barbara. Dr. Zhao joined CSU faculty in 2004. He is currently serving as the director of the Master of Science in Electrical

Engineering, and the Chair of the Graduate Program Committee in the Department of EECS, the ABET coordinator for the BS in Computer Science Program, and a member of the faculty senate at CSU. Dr. Zhao has authored a research monograph titled: "Building Dependable Distributed Systems" published by Scrivener Publishing, an imprint of John Wiley and Sons. Furthermore, Dr. Zhao published over 200 peer-reviewed papers on fault tolerant and dependable systems (three of them won the best paper award), computer vision and motion analysis, physics, and education. Dr. Zhao's research is supported in part by the US National Science Foundation, the US Department of Transportation, Ohio State Bureau of Workers' Compensation, and by Cleveland State University. Dr. Zhao has served on the organizing committee and the technical program committee for numerous international conferences. Dr. Zhao is an Associate Editor for IEEE Access, an Academic Editor for PeerJ Computer Science, and is a member of the editorial board for International Journal of Parallel Emergent and Distributed Systems, International Journal of Distributed Systems and Technologies, International Journal of Performability Engineering, International Journal of Handheld Computing Research. Dr. Zhao is a senior member of IEEE.

Function Approximation through an Efficient Neural Networks Method

Chaomin Luo[1], Zhuming Bi[2], and Wenbing Zhao[3]

Department of Electrical and Computer Engineering[1]
Mississippi State University, Mississippi State, MS 39762, USA
Department of Civil and Mechanical Engineering[2]
Purdue University Fort Wayne, Fort Wayne, IN 46805, USA
Department of Electrical Engineering and Computer Science[3]
Cleveland State University, Cleveland, OH 44115, USA

Abstract

Neural network system, a portion of artificial intelligence, is increasingly becoming prevalent nowadays. This paper presents a pedagogical study applied in a neural network field. The application of neural network models to function approximation is one of the latest developments in electrical engineering including robotics motion planning and navigation. It is, however, a challenging task to instruct on this topic in computational intelligence techniques course. In general, a function approximation issue aims to select a function among a well-defined class that closely matches a target function in a task-specific manner, which has a large number of applications in engineering such as, face recognition, image classification, and robotics navigation and motion planning. In this paper, we present how we follow the project-based and *divide-and-conquer* pedagogies to help students design, implement, debug, and operate an efficient neural networks model for function approximation. The effectiveness of the neural network model for function approximation is evaluated through various milestone assignments, milestone reports, presentations, and other activities. Teaching and learning strategies of this neural network model methodology for function approximation were validated by learning outcomes of this course by analysis of the neural networks model for function approximation project.

1. Introduction

In higher education, especially, in engineering education, all the effort of educators is focused on educating students to be qualified for their future professions. To achieve these learning outcomes in engineering education, a variety of pedagogical considerations have been implemented and experimented. There have been a large number of pedagogical efforts proposed and implemented, such as inquiry-based learning (Behrouzi and Kuchma, 2016), project-based learning (Khorbotly, 2015; Wang *et. al*., 2017; Zhao *et. al*., 2017; Luo, 2015), cooperative learning (Akili, 2012), active learning (Luo, 2015; Akili, 2014; Oliveira, 2015), divide-and-conquer learning (Kos and Miller, 2017; Sullivan-Green, *et. al.*, 2017), student-centered learning (Weimer 2002; Grigg and Stephan, 2018), and problem-based method (Oliveira, 2015), etc.

Behrouzi and Kuchma addressed an inquiry-based learning pedagogy used in a freshman civil and structural engineering curriculum with an equipment-light laboratory course (Behrouzi and Kuchma, 2016). The project-based learning approach is one of the innovative methods promoted in engineering education. Khorbotly developed a computer vision curriculum in the undergraduate electrical engineering program using a project-based learning pedagogy. Some issues implemented in project-based approach are addressed (Khorbotly, 2015). Luo presented an *on-going* multiple-project-based pedagogy in electrical and computer engineering program. In this course, a sequence of well-prepared projects was assigned to students to cover various topics to help student learning for enhancement of research skills (Luo, 2015).

Active-based learning is a learning protocol, in which teaching strives to involve students in the learning process more directly than in other methods (Luo, 2015). It emphasizes learning without the burden of assignments and without assessment through intimidating exams and tests unlike a traditional classroom environment (Ganago *et. al.*, 2016). Oliveira adopted active learning approaches to encourage active learning and engagement among students in face-to-face electrical engineering technology courses. The assessment results demonstrated that the active learning strategies have successfully met the teaching requirements (Oliveira, 2015, Luo, *et. al.*, 2017).

Cooperative learning activities promote peer interaction and assist the development of engineering course in terms of better learning of concepts and contents. Akili developed a cooperative learning method in a large-scale engineering course, in which the cooperative learning has been proven to be effective (Akili, 2012). Kos and Miller utilized a divide-and-conquer learning scheme to teach a large freshmen engineering course, by decomposing the course work into two types of assignments, weekly homework and a final report (Kos and Miller, 2017). Grigg and Stephan used a student-centered learning pedagogy in activities for large-enrollment undergraduate programs setting to foster a quality learning experience for engineering students through delivering foundational knowledge and facilitating skills development (Grigg and Stephan, 2018).

Some engineering educators incorporated two or more pedagogical strategies for advancement of education quality. For instance, Khorbotly combined a project-based pedagogy and a lecture-base teaching method that better teach a computer vision class in the electrical and computer engineering program (Khorbotly, 2015). Oliveira implemented active learning, and cooperative learning associated with a problem-based learning protocol in electrical engineering technology hands-on courses. It serves as evidence for the effectiveness of integrated strategies of active learning with other pedagogies (Oliveira, 2015).

This paper aims to renew the call for deployment of better and more effective instructional strategies in the computational intelligence techniques class, stressing on *integration* of project-based learning and *divide-and-conquer* learning practices as a viable alternative to the traditional talk-and-chalk setting that has gripped the engineering education. This paper presents a pedagogical study applied in a neural network field. We present how we follow the pedagogies of *integration* of project-based learning and *divide-and-conquer* learning to help students design, implement, debug, and operate an efficient neural networks model method for function approximation. A back propagation (BP) neural networks model is taught in the classroom with the MATLAB source code provided to students. The students are required to revise and modify the source code based on their understanding, therefore apply the back propagation neural networks model for the function approximation.

Specifically, in our computational intelligence techniques course for senior and graduate students, we provide a back propagation neural networks model with its source code as a sample for students to dissect and examine how the neural networks model for function approximation works. We then work together with students in revision and modification of the source code for purpose of function approximation. With the gradual withdrawal of the instructor in a series of projects assigned to students on the application of back propagation neural networks model to function approximation in robot path planning, students gain more independence and therefore are capable of fulfilling the projects fully by themselves in the end. A set of projects are assigned to students to perform the BP neural networks model for the function approximation. An assessment approach is applied to determine how comfortable students are with neural network concepts before and after the project. In addition, feedback was solicited after each milestone to gather feedback from students about implementation, development and application of the models into function approximation.

A mechanism that was very widespread in our data was one we called the "project-based" mechanism applicable in our computational intelligence techniques course. Project-based teaching and learning has been a major baseline of research and practice in engineering education due to engineering profession's particular requirement to link classroom learning to the actual ability needed in the professional world, especially, some topics that need to be taught in a divide-and-conquer mechanism. The divide-and-conquer learning in engineering education functions by recursively breaking down a project into more sub-projects of the same or related type, until these become simple and acceptable enough to be solved directly (Kos and Miller, 2017). The solutions to the sub-projects are then combined to give a solution to the original problem. The divide-and-conquer learning breaks the traditional lecture dominant pattern to improve the learning quality.

Teaching and learning strategies by the neural network model methodology were verified by the learning outcomes of this course by analysis of the function approximation project. All these data, together with those from the official course evaluation system, point to the effectiveness of the neural network model approach and high learning quality as a result of our project-based and *divide-and-conquer* pedagogies. The effectiveness of the neural network model for function approximation is evaluated through various milestone assignments, lab reports, presentations, and other activities, when the project-based learning and *divide-and-conquer* learning protocols are deployed. This paper shows how *integration* of project-based learning and *divide-and-conquer* learning scheme can advance academic success, quality of learning, and exploration toward the college experience.

2. Description of the Course and Project Design

The course of ELEE4400/5400 computational intelligence techniques is a Technical Elective course for electrical engineering, computer engineering, and robotics and mechatronic systems engineering students. It delivers two 75-minute lectures per week. With the advance of increasingly faster computing hardware and cheaper memory chips, computational intelligence, a part of artificial intelligence, a relatively new area of application in engineering, is becoming increasingly important in many engineering and non-engineering disciplines including robotics. This course addresses basic structures in computational intelligence techniques including neural networks, bio-inspired systems, genetic algorithms and swarm intelligence. As this course covers a number of useful topics, each topic is discussed deeply. It is crucial to emphasize that this course is different from the traditional theoretical-based engineering course. Some vital topics that benefit for engineering applications are selected from computational intelligence techniques.

The main focus is on teaching students to perform engineering projects to meet the required specifications while applying computational intelligence techniques. Neural networks have drawn remarkable attention from the machine learning, and computational intelligence research community. Due to their recent empirical successes, particularly, neural networks are used to build sophisticated systems in a variety of applications such as speech recognition, image recognition, and robot navigation and others. This will benefit for undergraduate seniors of their capstone project, such as, design and simulation of engineering systems modeled with computational intelligence techniques like neural networks, genetic algorithms, fuzzy logic and swarm intelligence techniques. Once graduate students have completed this course, it will benefit for them to have broader views. It will help graduate students to be ready to take graduate courses positively such as machine learning, robotics and automation, and arterial intelligence. Some topics such as neural networks, genetic algorithms, fuzzy logic and swarm intelligence will have influence on their research projects as well.

Function approximation is a fundamental issue in a vast majority of real-world applications. It aims to find the underlying relationship from a given input and output data. The requirements of this on-going project are described as follows.

- According to the given MATLAB code, students are required to draw BOTH a flow-chat and pseudo-code to explain BP neural networks for function approximation.
- Students are suggested to carry out the Function Approximation, by revising the provided MATLAB code, for y = cos(2piX1)*cos(2piX2). They are recommended to modify and play the parameters such as learning rate, number of hidden neurons and number of epochs to see what it will happen. They are required to record their results and discuss their observations; then plot the error figures, generated function and original function of y = cos(2piX1)*cos(2piX2).
- Students are encouraged to run the assigned MATLAB code before revising it to carry out the Function Approximation for some functions such as the following exponential function in equation (1):

$$\mathrm{Y} = \frac{1}{e^{x1} \times e^{x2}} \qquad (1)$$

- Students attempt to modify and play the parameters such as learning rate, number of hidden neurons and number of epochs to see what it will happen. They record the results and discuss their observations. Students must plot the generated function and original function of exponential function.
- Students should utilize some functions for function approximation to make the robot trajectory illustrated in Fig. 1 from S (2,2) to T (27,27) in blue circles. Student should clearly write their function mathematically in the milestone reports.

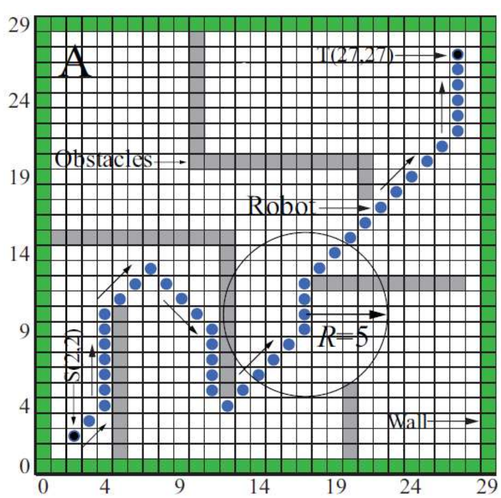
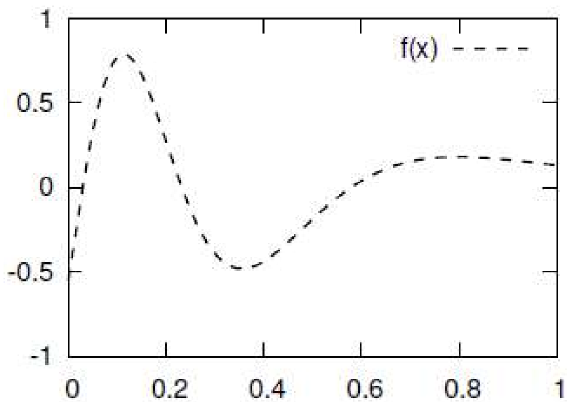


Fig. 1 Illustration of a function approximation for a robot navigation case

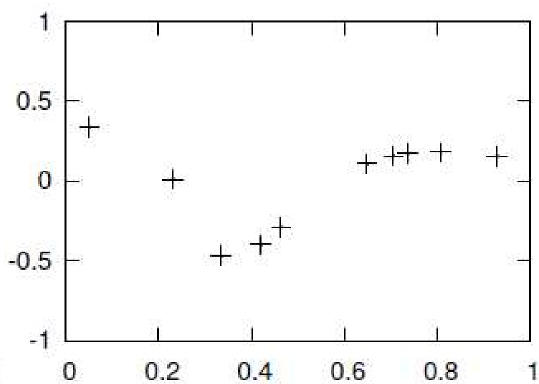
3. The Project Description and Consideration

3.1 Function Approximation

In general, a function approximation issue seeks to select a function among a well-defined class that closely approximates a target function in a task-specific mode. The need for function approximations arises in many real-world applications such as speech recognition, robot navigation, and the need for an approximating function often arises. A project-based methodology is utilized in the sense that students' all learning activities from one of the topics in computational intelligence design, construction, implementation, to testing are anchored in one semester-long design project. Numerous results have demonstrated the universal approximation property of neural networks in approximations of different function classes (see Fig 2). One of the major applications of BP neural networks is to approximate a function of several variables that can be used for robot navigation. Neural network training can be regarded as function approximation (see an example in Fig. 2).



(A)



(B)

Fig. 2 Illustration of a function approximation (A) The unknown underlying function; (B) Random samples with function approximate (*redrawn from Stalph, 2014*)

2 Back Propagation Neural Networks Model

In this well-prepared project, we teach students back propagation (BP) neural network algorithm and its applications in function approximation through a sequence of milestone-driven sub-projects by the *divide-and-conquer* learning scheme. BP neural networks are often utilized for statistical analysis, function approximation, and data modelling, in which their role is perceived as an alternative to standard nonlinear regression or cluster analysis techniques (Gurney 1997), in many real-world applications including image and speech recognition, textual character recognition, and financial prediction. This type of issue falls within the domain of classical artificial intelligence (AI). The BP neural network, in which an error is calculated at the output and passed backwards throughout the network's layers, is a sort of multi-layer feed forward network. It is usually used to train deep neural networks. A BP neural network has adaptive learning abilities to estimate sampled functions, represent these samples, encode structural knowledge, and inference inputs to outputs through association. In terms of learning, back propagation is usually used by the gradient descent optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function. Its main strength lies in its (sufficiently large number of) hidden units, thus a large number of interconnections. Some fundamental principle is described as follows that has been encompassed and taught in the class and enforced in the project.

The project with BP neural networks for function approximation covers the following items.

- Introduction of BP neural networks
- Distribution and exercise of sample MATLAB code of BP neural networks
- Introduction of function approximation
- Analysis of BP neural networks to function approximation based on given code
- Introduction of parameters of the BP algorithms
- Introduction of variation of functions of BP neural networks
- Data-driven BP neural networks algorithm for function approximation
- Introduction of revision of BP neural networks algorithm for function approximation
- Introduction of revision of BP neural networks algorithm for robot navigation

In the BP neural networks algorithm, the output is described as equation (2):

$$o = f(net) = f(\sum_{i=1}^{n} w_i \cdot x_i) \qquad (2)$$

The activation function is selected as step function as in equation (3), or sigmoid function as in equation (4), respectively (such as Fig. 3).

$$f(x) := \operatorname{sgn}(x) = \begin{cases} 1 & if \quad x \geq 0 \\ -1 & if \quad x < 0 \end{cases} \qquad (3)$$

$$f(x) := \varphi(x) = \frac{1}{1+e^{(-ax)}} \qquad (4)$$

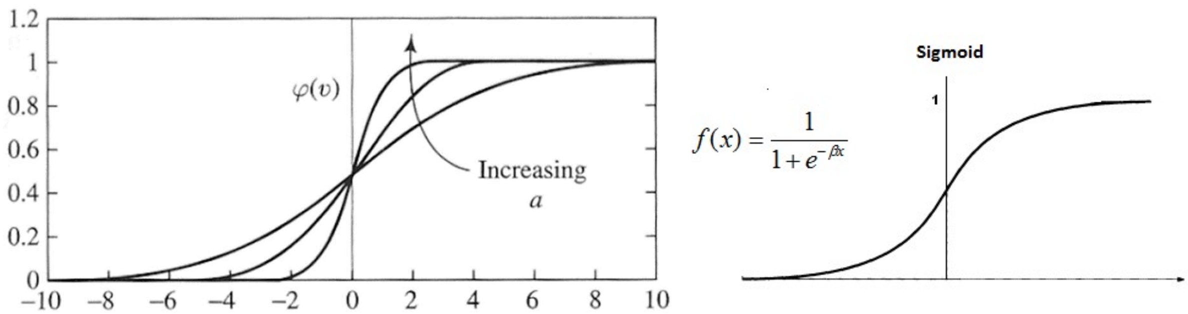


Fig. 3 sigmoid function used as the activation function in the BP neural networks

A two-layer BP neural network is considered, in which the weights w_{ij} in any in any feed-forward network to learn a training set of input output pairs $\{\boldsymbol{x}_d, \boldsymbol{t}_d\}$. Given the pattern $\boldsymbol{x}_d$ the hidden unit j receives a *net* input as equation (4) and produces the output as equation (5).

$$net_j^d = \sum_{k=1}^{5} w_{jk} x_k^d \qquad (5)$$

$$V_j^d = f(net_j^d) = f(\sum_{k=1}^{5} w_{jk} x_k^d) \qquad (6)$$

Output unit i thus receives the following,

$$net_i^d = \sum_{j=1}^{3} W_{ij} V_j^d = \sum_{j=1}^{3} (W_{ij} \cdot f(\sum_{k=1}^{5} w_{jk} x_k^d)) \qquad (7)$$

Therefore, it produces the final output

$$o_i^d = f(net_i^d) = f(\sum_{j=1}^{3} W_{ij} V_j^d) = f(\sum_{j=1}^{3} (W_{ij} \cdot f(\sum_{k=1}^{5} w_{jk} x_k^d))) \qquad (8)$$

Thus, for l outputs and m input output pairs $\{\boldsymbol{x}_d, \boldsymbol{t}_d\}$, the error function is following.

$$E[\vec{w}] = \frac{1}{2}\sum_{d=1}^{m}\sum_{i=1}^{l}(t_i^d - o_i^d)^2 \qquad (9)$$

The weights are updated as follows. $\Delta w_i = \eta(t-o)x_i \qquad (10)$

Students carried out the sub-projects to gain the knowledge of the BP neural networks for function approximation by running the code, tuning parameters and understanding the model. Students need to record their results and discuss their observations in milestones and discussed with the instructors their progress such as Figs. 4 and 5. Students must plot the generated function and original function such as Y=1/(e^x1×e^x2).

4. The Project based Pedagogy Integrated with Self-Assessment Method

In light of divide-and-conquer learning protocol, the ultimate project is divided into a sequence of small sub-projects to create rhythm and allow space for self-regulation to be acted. Integrated with the divide-and-conquer learning scheme might be a unique feature of the pedagogy in this study compared to other studies of project-based teaching and learning. The idea of BP neural networks for function approximation with applications into a mobile robot navigation is illustrated in Fig. 6. A variety of sub-projects, briefly presented in Table 1, have synergistic inter-connections. The session of Sub-Project 1 (SP_1) is also an orientation session informing students that each sub-project is a stepping milestone for the next in the entire series that culminate in the final project.

Each sub-project shown in Table 1 with student performance was used to evaluate progress students made through some activities including a written sub-project report and a sub-project interview with the instructor. For instance, The BP algorithm is a sensible approach for dividing the contribution of each weight. Hands-on project enables students to better understand that BP neural networks of learning principles have two categories: hidden layers and gradients. By sub-projects SP_6 and SP_7, students understand that there are two differences for the updating rule: (1) The activation of the hidden unit is used instead of activation of the input value; (2) The rule contains a term for the gradient of the activation function.

Some questions are prepared for students while they carry out the sub-projects. For example, how many hidden units in the layer in the BP neural networks for function approximation. Through the implementation of sub-projects, these students realize if hidden units are excessively few, the networks would not learn well. After SP_3, through tuning the parameters and revising the code, students realize that BP neural network is conceptually simple, and the global error is backward propagated to network nodes, weights are modified proportional to their contribution. This paper sheds light on project definition, sub-project division, project activities, self-regulation and assessment to stimulate interaction between the instructor and students.

Table 1 The descriptions of the sub-projects

Sub-Projects (SP)	Description	Average Performance (%)
SP_1: Distribution and exercise of sample MATLAB	○ Students are initially assigned a set of MATLAB source code, in which it implements back propagation neural networks for function approximation.	

code	○ Students are required to understand the MATLAB code fully and run it. ○ In this stage, instructor is in need of developing a requirements definition, undertaking a context analysis, and exploring design constraints. ○ The project plan is made. The overall design should be fulfilled.	86
SP_2: Analysis of BP neural networks to function approximation based on given MATLAB code	○ Students are supposed to prepare flowchart of the BP neural networks algorithm using block diagrams and schematics. ○ Students are encouraged to construct code of the BP neural networks algorithm for function approximation. ○ Students are supported to play weights to have performance analysis of BP neural networks.	87
SP_3: Organization of parameters of the BP algorithms	○ Students are taught to tune the parameters of BP neural networks model. ○ Students are encouraged to change parameters and evaluate their impact on performance of BP neural networks model.	89
SP_4: Variation of functions	○ Students are required to substitute the given function by some functions to be approximated. ○ Students are taught to tune parameters of BP neural networks model to fit in new functions to be approximated. ○ Students are encouraged to evaluate the performance of BP neural networks model in term of various parameters of BP neural networks model with a variety of functions.	90
SP_5: Data-driven BP neural networks algorithm for function approximation	○ Students are assigned some data (set) with input-output to implement the BP neural networks model to function approximation. ○ Students are distributed some robot trajectory samples ○ Students try to run and debug the BP neural networks model code to fit it in the robot path planning and navigation	93
SP_6: Entire model test and debugging	○ Students revise, test and debug BP neural networks model code to ensure the function approximation to be suited for the robot smoothing trajectory. ○ Students practice the test of two differences for the updating rule of BP neural networks.	94
SP_7: Revision of BP neural networks algorithm for robot navigation	○ Students perform more revisions of BP neural networks model code to implement robot smoother robot trajectory on simulation studies. ○ Students practice updating interior weights of the BP neural networks algorithm for robot navigation	89
SP_8: Further implementation and improvement of	○ The design products from the previous sub-projects are integrated into the BP neural networks algorithm for robot navigation	91

Final: project demo, report, and presentation	o Students integrate all previous efforts into one design project to test its applicability and effectiveness for the robot navigation through the BP neural networks algorithm for function approximation. o Students are required to finalize, demonstrate, and the present the project. o Students need to turn in the deliverable before the deadline.	95

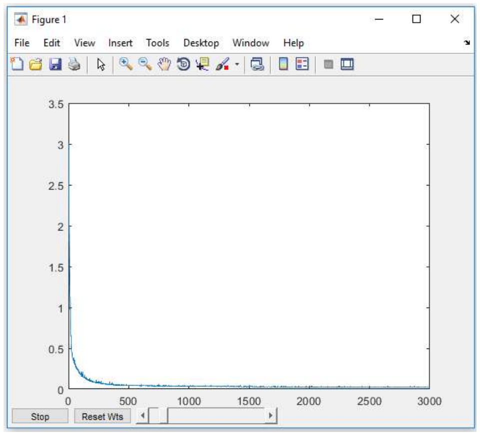
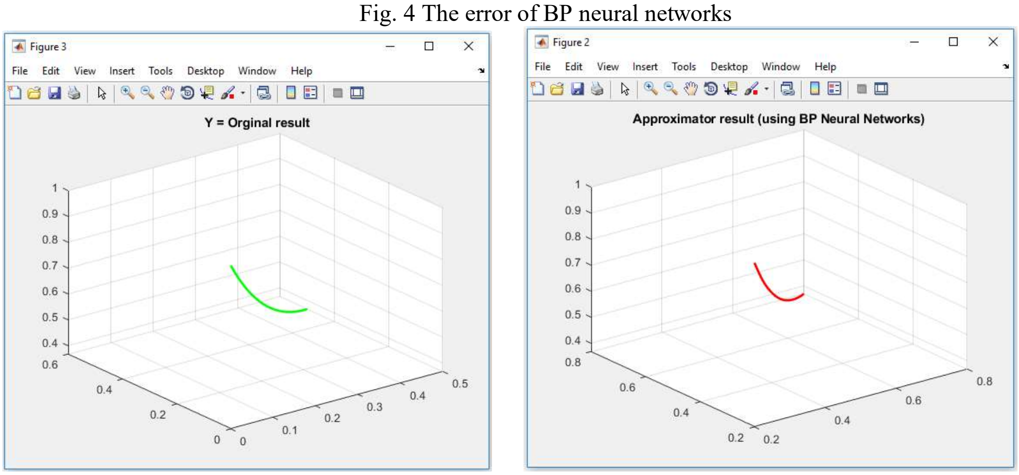


Fig. 4 The error of BP neural networks



(A) (B)

Fig. 5 Illustration of the function approximation (A) The orginal function; (B) The function approximation result

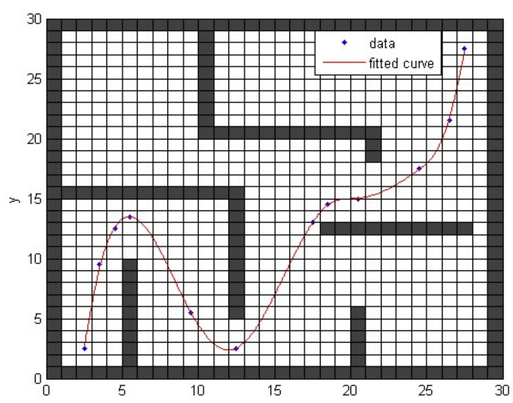


Fig. 6 Illustration of function approximation of robot navigation result

5. The Self-Assessments to Gauge Learning Outcomes

Aligned to ABET learning outcomes, the self-assessments are utilized for ABET assessment. These self-assessments are fulfilled before the end of the semester that can provide basis for instructors to improve the teaching and the course design, and more significantly, validate the learning quality and academic success. Students in this course respond to the six questions as in Table 2 (note that in the parentheses are the corresponding ABET outcomes).

Table 2 The questionnaire of students for assessment of education quality

Questions and Outcome		Survey			
		Strongly agree	Agree	Disagree	Strongly disagree
Q1	a	44.4%	55.6%	0%	0%
Q2	b	55.6%	44.4%	0%	0%
Q3	c	88.9%	11.1%	0%	0%
Q4	e	66.7%	33.3%	0%	0%
Q5	g	55.6%	44.4%	0%	0%
Q6	g	77.8%	22.2%	0%	0%

- Question 1 - "I can understand and use knowledge of mathematics including advanced topics such as differential and integral calculus, linear algebra, discrete math, and differential equations." (Outcome a: An ability to apply knowledge of mathematics, science, and engineering principles to electrical engineering; i.e. Knowledge of mathematics encompasses advanced topics typically including differential and integral calculus, linear algebra, complex variables, discrete math, and differential equations.)
- Question 2 - "I can apply formal engineering design methodology to perform the design, experiments and construction of the artificial intelligence projects based on experimental test data and interpretation, as well as to analyze and interpret data relating to artificial intelligence projects that resolve electrical system problems" (Outcome b: An ability to design and conduct experiments, as well as to analyze and interpret data relating to electrical systems.)
- Question 3 - "I can understand and design basic artificial neural network systems with assigned a sequence of projects such as feed-forward neural networks, and work to meet the final goals." (Outcome c: An ability to design electrical systems, components, or processes to meet desired needs).
- Question 4 - "I can understand structures of artificial neural networks and perform training for neural networks and profoundly understand some important learning rules such as Hebbian, Perceptron and Delta learning rules, and can also identify, formulate, and solve the issues raised in assigned artificial neural network projects" (Outcome e: An ability to identify, formulate, and solve electrical engineering problems.)
- Question 5 - "I have effective communication skills in the context of a collaborative, multi-disciplinary design activity in the project". (Outcome g: An ability to communicate effectively.)
- Question 6 - "I can create professional documentation in connection with the assignments and design project". (Outcome g: An ability to communicate effectively.)

The assessment questionnaire results are depicted in Table 2 and hence Fig. 7. As stated in Table 2, all students 'strongly agree' or 'agree' with the statements aligned to the ABET outcomes (a), (b), (c), (e) and (g). None of students disagree the statement in the learning outcomes. In light of the instructor's experience of teaching other courses, the percentages for 'strongly agree' are higher in this course, implying a possible effect of the pedagogies developed. Especially, on Questions 3 and 6, student percentages of 'strongly agree' are much higher at 88.9% and 77.8% than usual. Question 3 is tied to an ability to design electrical systems, components, or processes to meet desired needs, and Question 5 is related to an ability to communicate effectively, illustrated in Fig. 7 (Zhao, *et. al.*, 2017). In addition, in comparison with the instructor's previous experience teaching this course with a traditional project-based method (*i.e.* no sub-projects, no *divide-and-conquer* learning protocol and no interview sessions for reflection and adjustments), the percentages for 'strongly agree' and 'agree' in the current course are higher as well. It is inferred that the integrated version of the project-based pedagogy and a *divide-and-conquer* learning mechanism in this course is much more effective than previous models.

Results of learning outcomes and self-assessment indicate that this fused model of the project-based and divide-and-conquer based pedagogy is effective and efficient to enhance the learning quality. It has promoted their forethought, self-motivation, execution, exploration, discovery and learning of artificial intelligence techniques. This is an interesting phenomenon that not a single student answered "Disagree" or "Strongly Disagree". This is because of two reasons. Firstly, this group of students has a higher GPA. There are more A-level students in this group. Secondly, due to our pedagogies of the hybrid model, instructor took more efforts to help students. This would be an interesting engineering education research topic in the future.

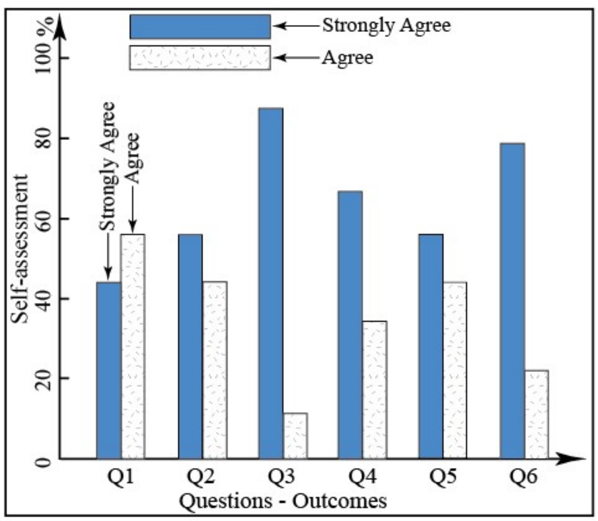


Fig. 7 Illustration of the self-assessment results

6. Conclusion

Our effort in approaching the instruction of modified version of the project-based methodology function approximation through an efficient neural network has been reported in this paper. The project-based and *divide-and-conquer* based pedagogies have been developed to assist students in designing, implementing, debugging, and operating an efficient neural networks model method for function approximation. The effectiveness of the neural network model for function approximation was evaluated through various milestone assignments, milestone reports, presentations, and other activities. Results from students' self-assessment questionnaire provided further evidence that the project-based integrated pedagogy is effective in achieving the learning outcomes designed for this computational intelligence techniques course. Comparison studies of same course taught in two different years using different pedagogies will be an interesting topic in the future research.

References

1. A. Behrouzi, and D. Kuchma (2016, June), Inquiry-Based Learning to Explore the Design of the Built Environment Paper, *ASEE* (American Society for Engineering Education) *Annual Conference & Exposition*, New Orleans, LA, USA. 10.18260/p.25725.
2. S. Khorbotly (2015, June), A Project-based Learning Approach to Teaching Computer Vision at the Undergraduate Level, *2015 ASEE Annual Conference & Exposition*, Seattle, Washington, USA, June 14-17, 2015.
3. J. Wang, C. Luo, W. Zhao, and X. Li (2017, June), Empowering Students with Self-Regulation in a Project-Based Embedded Systems Course, *124th ASEE Annual Conference & Exposition* (ASEE'2017), June 25 - 28, 2017, Columbus, Ohio, USA.
4. W. Zhao, X. Luo, C. Luo, and Y. Peng (2017, June), Design and Implementation of Project-Based Courses on Cutting-Edge Computer Technologies, *124th ASEE Annual Conference & Exposition* (ASEE'2017), June 25 - 28, 2017, Columbus, Ohio, USA.
5. C. Luo (2015, June), Enhancing the Quality of Student Research by an On-going Multiple-Project-Based Course, *122nd ASEE Annual Conference & Exposition* (ASEE'2015), Seattle, WA, USA, June 14-17, 2015.
6. W. Akili (2012, June), Cooperative Learning Protocols: Varieties and Obstacles, *2012 ASEE Annual Conference & Exposition*, San Antonio, Texas. https://peer.asee.org/21106
7. C. Luo (2015, June), Hands-On Active Learning via Development of FPGA-based Intelligent Microwave Oven Controller, *122nd ASEE Annual Conference & Exposition* (ASEE'2015), Seattle, WA, USA, June 14-17, 2015.
8. A. M. Oliveira (2015, June), Simple Ways so Facilitate Active Learning in Hands-on Electrical Engineering Technology Courses, *2015 ASEE Annual Conference & Exposition*, Seattle, Washington, USA, June 14-17, 2015.
9. W. Akili (2014, June), On Implementation of Classroom-Based Pedagogies of Engagement: Relevant Measures and General Outcomes, *2014 ASEE Annual Conference & Exposition*, Indianapolis, Indiana, USA. https://peer.asee.org/22882
10. L. E. Sullivan-Green, R. Mathur, and A. H. Feinstein (2017, June), Flipping STEM Classrooms Collaboratively Across Campuses in California, *2017 ASEE Annual Conference & Exposition*, Columbus, Ohio, USA. https://peer.asee.org/28370
11. S. J. Grigg, and E. A. Stephan (2018, June), (PREP)ARE: A Student-centered Approach to Provide Scaffolding in a Flipped Classroom Environment, *2018 ASEE Annual Conference & Exposition*, Salt Lake City, Utah, USA. https://peer.asee.org/29651
12. C. Luo, J. Wang, W. Zhao, and L. Wang (2017, June), Multi-Lab-Driven Learning Method Used for Robotics ROS Study, *124th ASEE Annual Conference & Exposition* (ASEE'2017), June 25 - 28, 2017, Columbus, Ohio, USA.
13. W. Zhao, X. Liu, C. Luo, and X. Luo (2017, June), Enhancing Communication with Students with a Teaching Method Based on Topical Guide Objectives, *124th ASEE Annual Conference & Exposition (ASEE'2017)*, June 25 - 28, 2017, Columbus, Ohio, USA.
14. K. Gurney (1997), An Introduction to Neural Networks 1st Edition, CRC Press, August 7, 1997), ISBN-13: 978-1857285031.
15. Weimer, M. (2002). Learner-centered Teaching: Five Key Changes to Practice. San Francisco: Jossey-Bass Publishers, 2002.
16. A. Ganago, H. J. Kim, J. A. Kotrba, and M. Rasouli (2016, June), Teaching Electrical Engineering to non-EE Majors in a Flipped or Blended Classroom, 2016 ASEE Annual Conference & Exposition, June 2016, New Orleans, LA, USA.