TEACHING SIGNAL PROCESSING CONCEPTS TO DIGITAL NATIVES

Gregory A. Krudysz and James H. McClellan

Georgia Institute of Technology
School of Electrical Engineering
Atlanta, GA, USA

ABSTRACT

Teaching is increasingly migrating to digital platforms, where relevant content can be quickly accessed and explored in small chunks within the appropriate context. A variety of on-line systems have been developed to cater to educational experience that is predominantly digital, contextually accurate, and personalized to learner's conceptual understanding and needs. In this paper, we provide an update on our continuing efforts at Georgia Tech to transform an undergraduate signal processing course into a data-mining teaching environment that structures student learning through concept driven content and personalized programmed instruction.

***Index Terms*—** tutoring system, automated teaching, on-line labs, signal processing, data-mining, digital natives

1. INTRODUCTION

Learning is most effective when learners take charge of their own educational experience by immersing themselves in the content and context of the topic that they aspire to acquire. It is the role of the teacher to guide the student into the topic through structured conceptual development that offers presentation of the material in digestible chunks, in the form of progressive reinforcement, sequenced at the right pace, and with timely feedback on their understanding and performance. Through the learning experience, the student progresses from the role of a *spectator* of content presentation, to that of a skilled *participant* of knowledge, where educational goals are formulated in pursuit of higher-order concepts, and learning is enhanced through deliberate practice and appropriate application of the newly acquired knowledge.

Today's teaching environment has been transformed by technology – as it always has been, however only recently have we reached the possibility of a full scale programmed instruction, where education incessantly is becoming automated to best adapt to *just-in time* learner's needs, tasks, problems to be solved, and applications to be understood and skillfully utilized.

The digital revolution has brought instantaneous access to vast quantities of information by connecting individuals to ideas, knowledge, and organizations. The incoming cohort of STEM students embarking on their engineering education has been described by technological demographers as the *net-gneration*, and also as – *digital natives*. These students have grown up immersed in the internet-based technology, and its network effects, with 95% of *generation Z* high-school students owning a smartphone. According to market studies, half of *genZ* students spend over 5 hours a day interacting with their electronic device, with 26% of these students spending over 10 hours each day, and 20+ hours a month are spent by engaging in video content [1]. With ubiquitous connectivity, data feeds, and pervasive interaction, *digital natives* have grown up with expectations of structured presentations and instant feedback.

With the ongoing shifts in the traditional employment in farming (1%) and manufacturing (12%) in decline [2], and with the advent of service-based *information* economy, an estimated 68% of current high-school graduates are expected to pursue college education, and 58% are projected to graduate with a bachelor's degree within 6 years of enrollment [3]. This largest generation of college bound students, comprising 23% of the total U.S. population, challenges institutions of higher learning to offer high-quality education that is both technologically engaging and highly personalized, yet also cost-effective.

2. TEACHING MACHINES AND PROGRAMMED LEARNING

Efforts to improve education on a large scale go back to the prediction of programmed instruction through the invention of a mechanical textbook by the education researcher Edward Thorndike who in 1912 opined: "If by a miracle of mechanical ingenuity, a book could be arranged that only to him who had done what was directed on page one would page two become visible and so on, much that now requires personal instruction could be managed by print" [4]. Only a year later, in 1913, the first patent for an implementation of an *educational appliance* designed in principle to measure learner's progress through *testing* of *arithmetic* and *writing* skills, was awarded to psychologist Herbert Atkins, and set off a century-long quest to improve learning on a mass scale.

In 1950's, with the emergence of the electronic medium, the *ideal teacher* was already defined in terms of programmed learning codifying the *theory of teaching*. "Dissection of the teaching act occurs by defining programming concepts like *pacing*, *promoting*, *confirming*, and *reinforcing* more precisely, indicating their theoretical significance. Subsequently, these variables are used in describing what the teacher does and the consequences of this action" [5]. With the scientific emphasis on empirically based pedagogical model that is more objective and amenable to replication, there is an opportunity to codify explicitly *what the teacher does* and begin to teach the machines to *teach*.

2.1. ITS

At Georgia Tech, one of the foundational classes taken by the undergraduate students is the "Introduction to Signal Processing" course, which for many students is their first foray into an engineering subject consisting of structured mathematical expressions, a steady layering of hierarchical concepts, and with an overall strong emphasis on abstraction. This course is organized around concepts introduced in the "DSP First" textbook [6]. In additional to lectures and recitation sessions, the course consists of in-class *lab* exercises, where students have the opportunity to apply their knowledge of the recently acquired signal processing theory and demonstrate their understanding through practical implementations.

Our on-going work has centered on developing tools and resources that support signal processing teaching, and over the past few years we have embarked on the development of the Intelligent Tutoring System (ITS) [7]. The primary goal of this research project is to data-mine student activity in order to gain insight into *how students learn* and the way that technology can play a part in understanding and applying the most effective teaching methods, by providing data-driven guidance and algorithmic feedback to both students and instructors alike.

ITS is a web-based educational platform [8] designed to enhance student conceptual understanding through targeted questions and supporting educational content. The system organizes and integrates a collection of legacy content in the form of textbook chapters [9], lecture slides, problem exercises, labs, and MATLAB GUIs – developed over the years by signal processing instructors and graduate students. ITS aims to micro-test and reinforce student's conceptual knowledge through successive scheduled problem assignments. Since 2009, the system has been deployed as an on-line homework component within the "Introduction to Signal Processing" course (ECE-2026), having served over 1.1 million questions to over $4,500$ students, with more than 500 students working through its problems during each academic year.

Initially the system's dual objective was to streamline and encode instructional tasks associated with teaching by providing an on-line platform that allowed for automated assignment dissemination and grading. Students engage in self-study at their own pace and receive immediate feedback on their conceptual understanding and subject mastery. Instructors benefit from the system through real-time monitoring of student engagement, conceptual progression, and learning outcomes. Furthermore, this platform enables instructors to scientifically study the *educational process* by developing concept-centered educational tools specifically tailored to programmed instruction.

The current ITS version incorporates concept-based assignment mode consisting of parametric questions. The system provides a set of instructor tools to facilitate the creation of assignments by implementing an editable course assignment scheduler and question linker to each corresponding assignment. The *question editor* tools extend ITS capability with the addition of *clone* and *new* features, to allow instructors to clone existing questions and add new ones to the questions database via a template-structured interface. In addition, the system incorporates the course textbook [10], as a reference for the assignment mode, which allows the system to track user's study activity and concept searches. ITS has been used by students as an on-line supplement, mainly outside of their *in-class* activity; its pedagogy model within the ECE-2026 course and mode of operation is summarized in Table 1, and described in the first column labeled "ECE-2026".

		ECE-2026	ECE-2026 Labs
PEDAGOGY	Assignments	Assigned with overlap Released for 3-4 weeks	Assigned weekly Released for 12 hrs
	Question Bank	Work on many questions from a very large set	Work all problems from a small set
	Question Delivery	Random	Sequential
	Usage	Outside of class	During lab
CONTENT	Assignments	Based on textbook chapters	Based on labs
	Concepts	From textbook	From textbook
	Questions	Multiple-choice/Matching/Computed Pre-test & Post-test	Multiple-choice, Matching, Computed
	Solutions	Infer from feedback	
INSTRUCTOR	Add Content	Create new question Clone new question Add new assignment	
	Link Content	1. Link concepts to questions 2. Schedule assignments 3. Link questions to assignments	1. Link concepts to questions 2. Schedule labs 3. Link questions to labs
USER	Select mode	Assignment | Concept	Lab
	Interaction	1. Select released Assignment 2. ITS randomly selects a Question 3. (a) Skip question with no penalty 3. (b) Answer question: score 4. ITS feedback 4. (optional) rate Question difficulty	1. Select released Lab 2. ITS sequentially selects a Question 3. Answer question 4. ITS feedback: user selection 4. (optional) rate Question difficulty
GRADING		Correct	Attempted/Correct/Reviewed

Table 1. ECE-2026 ITS system design: *on-line* homework (left), and proposed *on-line* lab (right)

2.2. ITS Labs

Over the past year, our educational development efforts have centered on augmenting the utility of the ITS system by incorporating an *active classroom* design. The goal of this project is to formulate the engineering educational experience as a data-driven process that will allow for computerized personalized learning with algorithmic feedback-based instruction. ITS is also part of the Vertically Integrated Projects (VIP) [11] program that introduces undergraduate students to research through software development and data analysis. Much of our recent progress has been in part possible due to student led projects, through the VIP: Intelligent Tutoring System (VIP-ITS) design course.

Our on-going expansion of the ITS system is the incorporation of ECE-2026 *labs* within the ITS framework. Currently, we are working on developing the ECE-2026 *in-class lab* sessions within the ITS to replace the *paper and pencil* worksheet labs through the use of web-based applets that will allow for personalized interactive learning through concept-centric demos, guided set of instructional questions, and reinforced by incremental micro-testing.

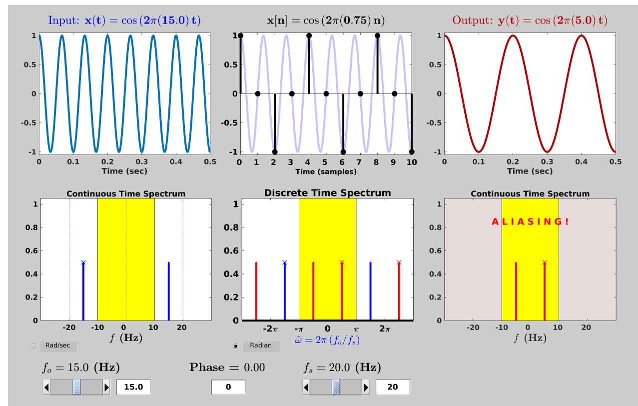
Currently, students taking the course are scheduled for *in-class* labs sessions, consisting of paper-based lab exercises, with a required *pre-lab* section to be completed before the *in-class* session, see Fig. 1 (a). Each lab is a mini-project guiding students through a MATLAB code-based implementation, of the concepts that they have learned, or an interactive educational exploration designed to develop deeper insight into concepts and their real-world applications. Since labs are one of the clearest examples of student active learning, there is an opportunity to bring ECE-2026 labs *on-line* and data-mine student activity in order to develop metrics of learner's conceptual understanding, necessary for the development of programmed instruction. Likewise, this flipped classroom setting, with web-based labs embedded within the ITS system and supported by on-line video narrations, would allow instructors to focus on in-person teaching, and for the *on-line* lab activity to be tracked, auto-graded, and analyzed for immediate feedback, with the opportunity for further scientific study.

Most of the labs make extensive use of the MATLAB GUIs, that have been developed by the ECE instructors and former students. Their educational efficacy stems from each GUIs inter-activity, where a concept can be demonstrated visually, often through simultaneous representations and multiple mathematical domains. For example, the *con2dis* MATLAB GUI explores the *sampling* and *aliasing* concepts in the "Sampling: A/D and D/A Aliasing" lab, see Fig. 1 (b). How a user interacts with a GUI and which parameters of the application have been selected within the context of the problem, is indicative whether a student has absorbed the lesson behind the demonstration and understood its underlying concepts. It is with this objective that the transition from MATLAB GUIs to web-based applets has been initiated through the VIP-ITS projects [6] – to set the stage for data mining efforts of the lab experience.

Recenlty, a number of web-based applets modeled on the existing MATLAB GUIs have been implemented through the VIP-ITS projects. For instance, the *filterdesign* MATLAB GUI used in the "FIR Filter Design" lab, has been written in

Part	Observe **and Justify** Verified:__________ Date/Time:__________
3.1(a)	The Nyquist rate for sampling with no aliasing is $f_s =$ Why?
3.1(b)	Within the $[-\pi, \pi]$ interval, $\hat{\omega} = \pm$
3.1(c)	Complex amp at positive $\hat{\omega}$.
3.1(d)	Within the $[-\pi, \pi]$ interval, $\hat{\omega} = \pm$
3.1(e)	Complex amp at positive $\hat{\omega}$.
3.1(f)	Three input frequencies and phases.
3.1(g)	Determine f_s to get a specific output frequency and phase.
3.2	Sketch spectrogram below. Explain. Verified:__________ Date/Time:__________

(a) "Sampling: A/D and D/A Aliasing" lab verification sheet.



(b) *con2dis* GUI used *in-class* lab to teach *sampling* and *aliasing* concepts

Fig. 1. ECE-2026 lab: (a) instructions, (b) MATLAB GUI

JavaScript using the JSXGraph library [12], and the prototype of the *on-line lab* version is shown in Fig. 2. Preliminary work on the *on-line lab* format combines the JSXGraph applet with the ITS database to simulate a lab interface capable of recording the *conceptual state* of each stduent's *in-lab* activity and their educational *context* as embodied by the supporting lab questions.

Due to its *active classroom* design, the on-line implementation of the ITS ECE-2026 lab has to account for user experience, whereby a list of relevant questions, along with instruction, is sequentially presented during short-time intervals. Also, the feedback presented to the user would include a list of supporting concepts, textbook table of contents and its related resources, and a set of questions mapped to key underlying concepts. The proposed ITS lab implementaiton is outlined in the Table 1, and its pedagogy details are described in the right column labeled "ECE-2026 Labs".

3. FUTURE WORK AND CONCLUSION

Over the next couple of semesters, we plan to deploy a trial-run of the *pre-lab* for the "Sampling: A/D and D/A Aliasing" lab within the signal processing lab course, then follow up with the deployment of an *in-class* lab, enhanced by video

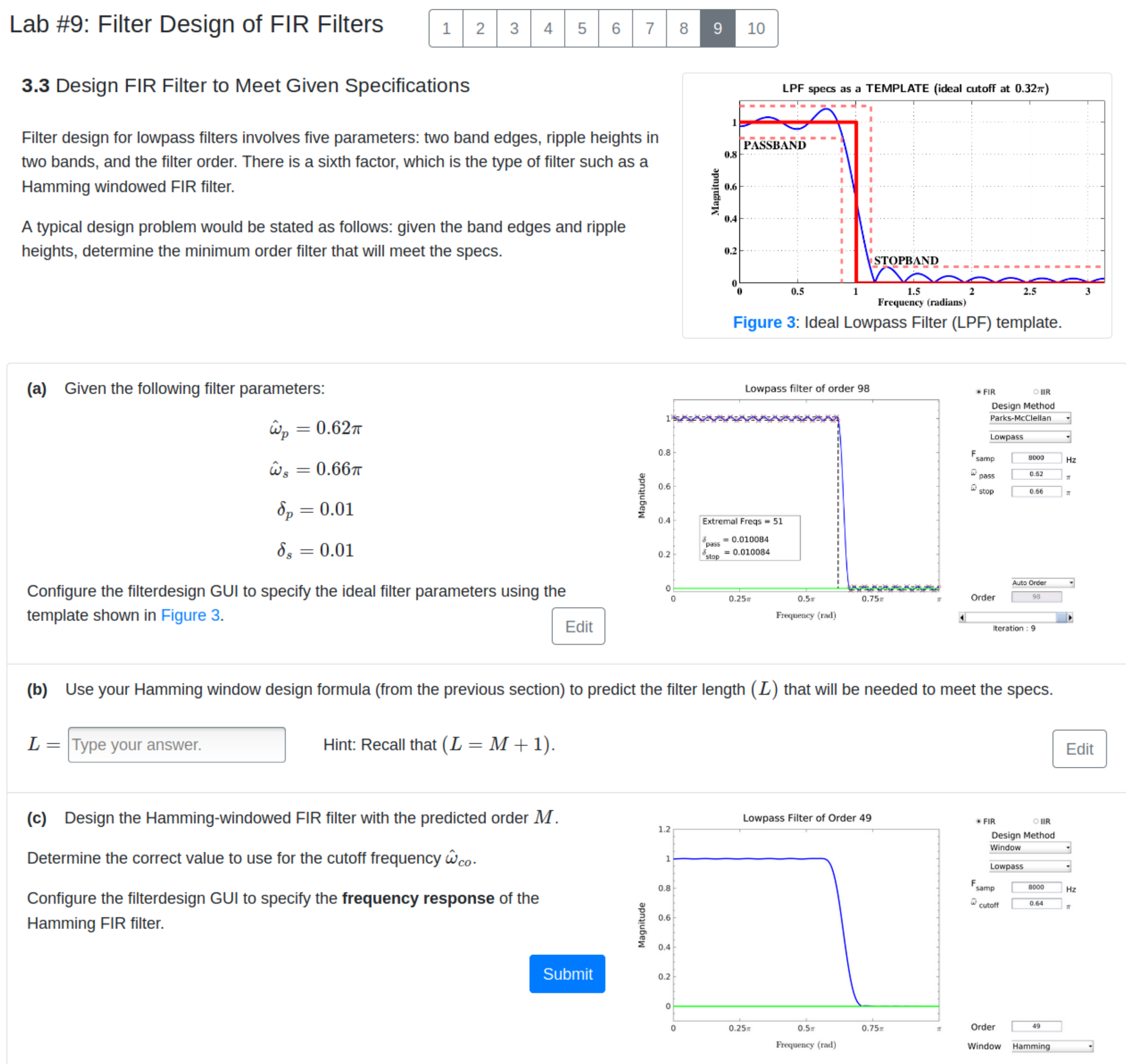


Fig. 2. ITS *on-line lab* prototype of the "FIR Filter Design" lab, with *in-lab* questions and *filterdesign* applet design.

narrations and a GUI component. Our recent progress on the ITS lab module is well suited for the task of implementing a self-contained web-based lab, and thus enhancing teaching of the signal processing course as an experimental data-driven programmed learning process.

4. REFERENCES

[1] Elli Denison Denise Villa, Jason Dorsey, "The state of gen z, 2018," Tech. Rep., The Center for Generational Kinetics, 2018.

[2] K. Dubina, A. Gensler, A. Lacey, and M. Toossi, "Projections overview and highlights," Tech. Rep., Monthly Labor Review, 201626, October 2017.

[3] S. Seppanen, "The millennial generation research review," *U.S. Chamber of Commerce National Chamber Foundation*, p. 2, November 2012.

[4] E.L. Thorndike, *Education: A First Book*, Macmillan Co., 1912, published 1923.

[5] edited by Alfred de Grazia and David A. Sohn, *Programs, Teachers, and Machines*, Bantam Books, 1964.

[6] J.H. McClellan, R. Schafer, and M. Yoder, *DSP First*, Pearson Education, 2015.

[7] G.A. Krudysz and J. H. McClellan, "Collaborative system for signal processing education," in *Proc. ICASSP-2011*, Prague, CZ, May 2011.

[8] G.A. Krudysz and J.H. McClellan, "Signal processing education through concept discovery and resource selection practice," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, CA, May 2013, pp. 26–31.

[9] J.H. McClellan, R.W. Schafer, and M.A. Yoder, *Signal Processing First*, Prentice Hall, 2003.

[10] G.A. Krudysz and J.H. McClellan, "Studyflow: Studying effective student-content interaction in signal processing education," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2017.

[11] Stephen Marshall, Edward Coyle, James V Krogmeier, Randal T. Abler, Amos Johnson, and Brian E. Gilchrist, "The vertically integrated projects (vip) program: leveraging faculty research interests to transform undergraduate stem education," 10 2014, Transforming Institutions: 21st Century Undergraduate STEM Education Conference ; Conference date: 23-10-2014 Through 24-10-2014.

[12] Michael Gerhauser, Bianca Valentin, and Alfred Wassermann, "Jsxgraph–dynamic mathematics with javascript," *International Journal for Technology in Mathematics Education*, vol. 17, no. 4, pp. 211–215, 2010.