

AC 2010-2231: AN INTERACTIVE LEARNING ENVIRONMENT FOR DSP

Shalin Mehta, Arizona State University

Jayaraman Jayaraman Thiagarajan, Arizona State University

Photini Spanias, Arizona State University

Photini Spanias, Ed.D. is a senior lecturer at CT&E of Arizona State University.

Karthikeyan Ramamurthy, Arizona State University

Robert Santucci, Arizona State University

Andreas Spanias, Arizona State University

Susan Haag, Arizona State University

Mahesh Banavar, Arizona State University

AN INTERACTIVE LEARNING ENVIRONMENT FOR DSP

Abstract

This paper presents a new version of the Java DSP (J-DSP) education software which encompasses a comprehensive Digital Signal Processing (DSP) learning environment. The goal is to create a one stop learning environment for students that wish to gain insight into DSP concepts. It combines html and Java based technologies to create an integrated and highly interactive simulation interface. The whole idea of this integrated environment revolves around a web-based DSP quiz. The quiz has many attractive features, which provide students an opportunity to gain a thorough understating of DSP theory. Different learning tools are combined with the interactive online quiz and are also synchronized with J-DSP simulations. Every quiz is accompanied with several helping features where users can access the instructor's notes and external links that contain simulations, lecture videos, and instructional J-DSP animations. This new environment also supports multiple J-DSP simulations, running concurrently. These simulations are completely independent with one another except for the certain predefined points at which two simulations can interact with each other and share their data. Assessments of the use of this new tool are also presented here.

Introduction

J-DSP is a java based DSP simulation tool that facilitates the understanding of discrete-time signal processing. J-DSP is platform independent and runs directly within a web browser. The simulation environment of J-DSP is interactive and helps in visualizing signal processing concepts. It provides users with a simple interface to build large and complex simulations using a simple drag and drop process. All the signal processing functions appear as blocks and can be connected graphically to establish a signal flow. J-DSP has a rich suite of signal processing functions that enable students to create interactive simulations online¹⁻¹⁰. J-DSP contains several basic as well as advanced DSP functions. Basic J-DSP functions include the Fast Fourier Transform (FFT), FIR and IIR filters, windows, arithmetic operations (addition, subtraction, multiplication and convolution), etc. Advanced functions include statistical DSP algorithms, speech processing functions, multirate signal processing functions, spectral analysis functions, etc. J-DSP also has few other versions which address the processing of different types of signals. For example, it has toolboxes for image processing^{11, 12}, control functions¹³, time-frequency signal processing¹⁴, communication related functions^{15, 16}, earth systems signal processing¹⁷ etc. A textbook that has J-DSP examples and exercises was also published¹⁸.

Even though the current version of J-DSP includes a broad array of DSP functions, there is a strong need to improve the students' overall learning and to enhance their online education experience. With the existing infrastructure and facilities, students have to logon to different web sites to access the online labs, lecture notes, lecture videos and J-DSP simulations. Moreover, these environments are independent of each other with no common link between them. This type of arrangement does not help in organizing the learning process and sometimes students spend a long time in finding the material rather than concentrating on learning the actual concepts. The

proposed interface attempts to address these issues by combining and connecting different learning environments together with the J-DSP simulations.

Figure 1 shows a high level overview of the proposed change with respect to the existing infrastructure. As shown, the proposed J-DSP interface consists of a number of discrete learning shell apart from the existing simulation environment. The learning environment could contain quizzes, online labs, animated demos, audio/video lectures, notes or web links that can reinforce specific DSP concepts.

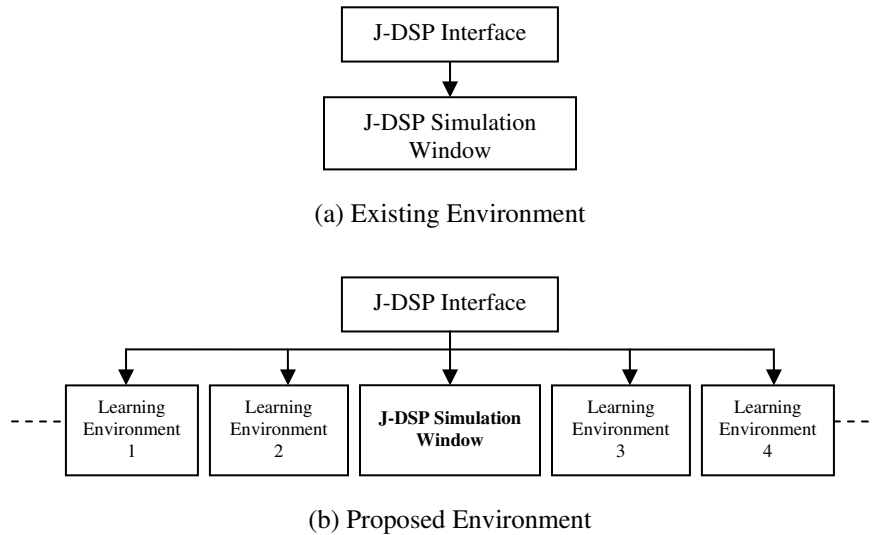


Figure 1. High level overview of the existing and the proposed infrastructure.

The Proposed Integrated Interface

The interface proposed here is built on the idea of combining a web-based DSP quiz and the J-DSP simulation window. Figure 2 shows the architectural view of the proposed interface. As can be seen from Figure 2, the on-line quiz is the connecting link between the diverse learning shells. After answering a quiz question, students can access multiple learning environments at the same time. The number of J-DSP simulation windows has also increased from one to three, to allow students to work on various concepts at the same time. The quiz is very interactive and gives students an opportunity to test their understanding of various topics. The learning environments are connected by means of questions posed in a quiz. All the three simulation areas and the web-based quiz are a part of a single browser window and reside in individual tabs.

Multiple Concurrent Simulations

The proposed interface has three independent J-DSP simulation windows. With the existing infrastructure, if one particular concept is to be simulated in a variety of ways, either a new J-DSP editor has to be opened for each simulation or simulations have to be performed one after

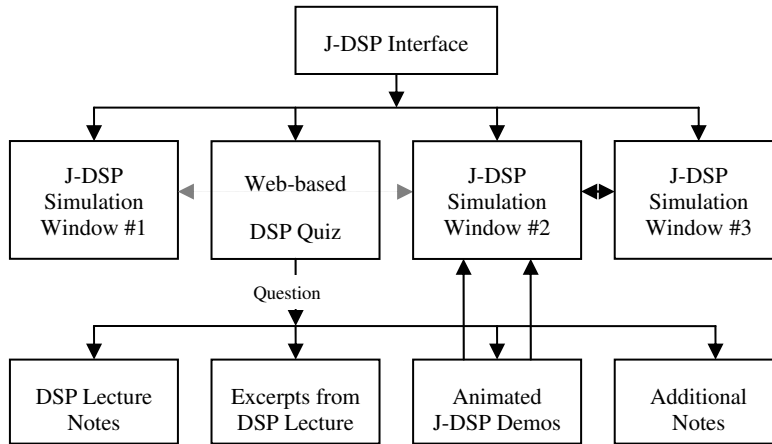


Figure 2. Architectural overview of the proposed infrastructure.

another. Both of the above mentioned methods are not elegant solutions to the problem of running multiple simulations simultaneously. Opening a new editor for each simulation consumes a lot of space and memory, and it also does not allow any communication between simulations as they are running in different browser windows. Creating simulations one by one does not help students in performing a clean comparison between the simulations and also does not allow them to work on different topics at the same time. The proposed J-DSP interface uses multiple tabs in a single window to address the needs of running multiple simulations and establishing a communication link between them. This functionality also helps in creating very large simulations which cannot be created in a single J-DSP window. Figure 3 and Figure 4 show the snapshots of the existing and the proposed J-DSP interface. The new environment has multiple tabs which contain three simulation windows and one quiz window. The buffer blocks are used to pass the output of one simulation as input to other simulations as shown in Figure 4.

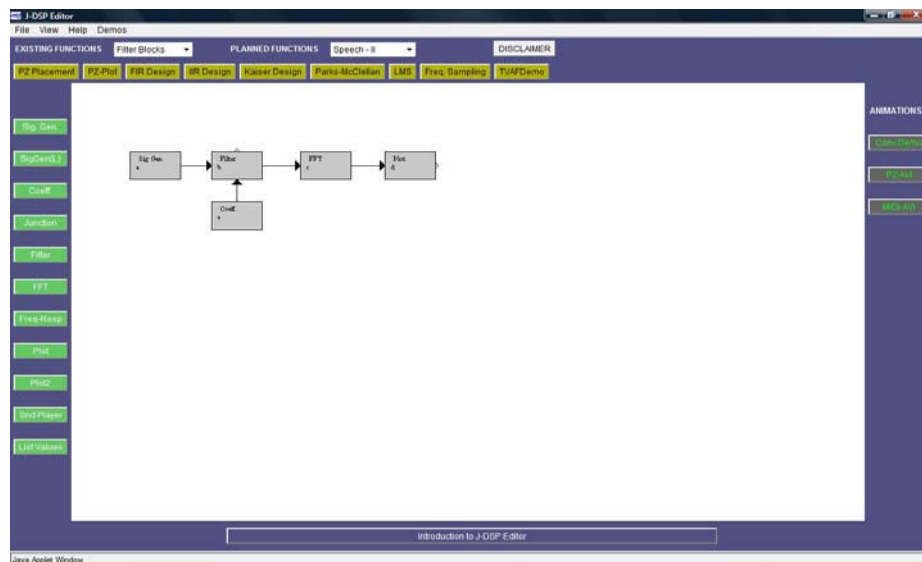
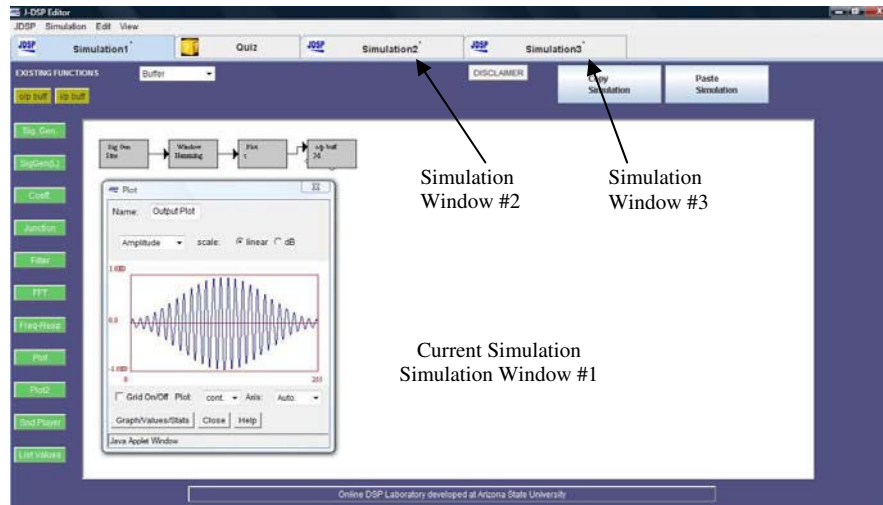
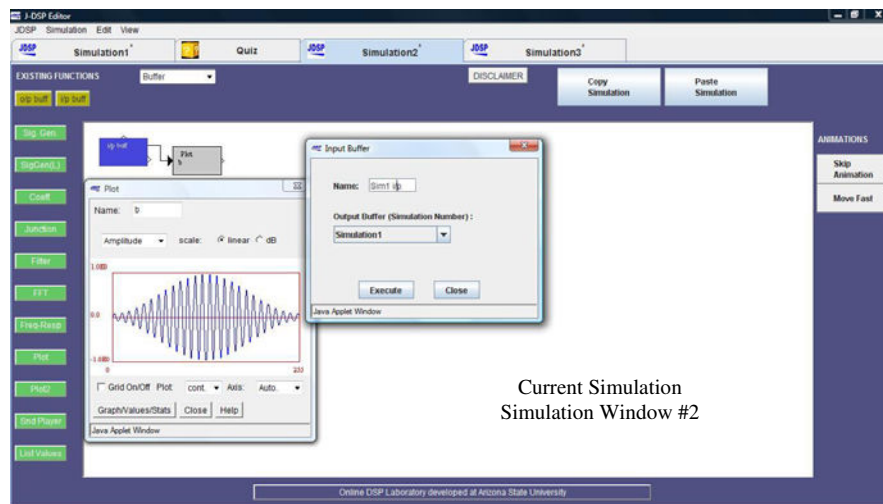


Figure 3. Current J-DSP interface.



(a) Output of Simulation1 captured in the output buffer.



(b) Input buffer of Simulation2 collecting signal from the output

Figure 4. Communication between two simulations using Input and Output buffers.

Web-based DSP quiz

The primary goal of this work is to create a single place for learning. The on-line DSP quiz explained in this section can be said to be the centerpiece of this new learning environment. The new interface makes sure that the students do not have to go to different locations to learn various concepts, as it combines the different learning environments. Figure 5 shows the quiz window with its components. After answering a question, students get access to the learning environments that not only cover the question but also can help them comprehend the entire topic on which the question is based. Once students attempt a question, they can go through the relevant excerpt from the lecture video to learn more about that subject. They can review the lecture slides or can read through the additional notes embedded in the quiz window itself. They can also have a look at the animated J-DSP simulation demo to visually understand the concept. The links to all of the above mentioned learning environments reside in the quiz window. These

links are initially disabled when the students are presented with the question, but they are enabled as soon as the students attempt the answer.

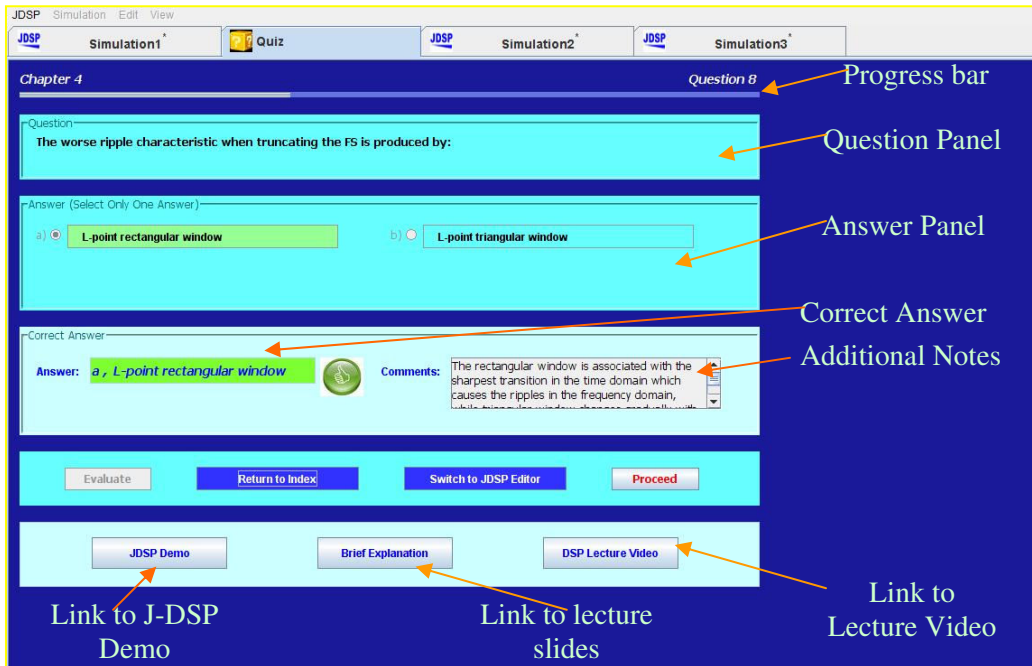


Figure 5. Quiz window in the Proposed J-DSP interface.

Quiz Features

The quiz has three types of questions. First are “filling the blanks” type of questions. Figure 6 shows a question-answer panel for this type of question.

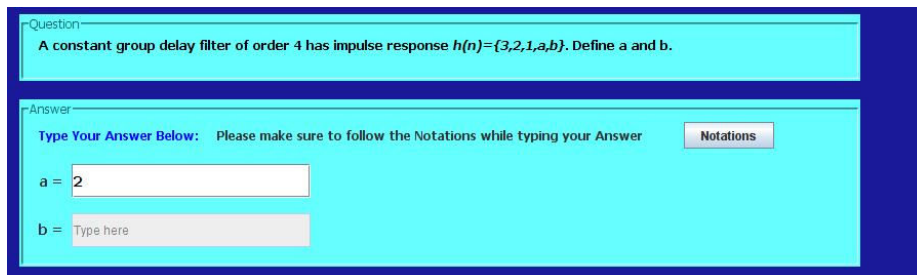


Figure 6. Fill the blanks type of quiz.

These types of questions are very difficult to correct by the automated program due to many possible ways of typing the correct answer. For example, averaging filter is a _____. The answer could be lowpass filter, LowPassFilter or Low-Pass Filter; all are the same and correct. In the same way $\exp(2) + 3$, $e^2 + 3$ or $3 + \exp(2)$ are the same. The students cannot type the special symbols as well, as there is no equation editor in the answer field. For example, if the correct answer is $\delta(n)$, they might type d(n) or delta(n). This makes the parsing and correction of the answer entered by the user a challenging task. To make the things simpler and uniform, users

are provided with the list of symbols which are to be strictly followed while entering the answer. Figure 7 shows a part of the notations dialog.

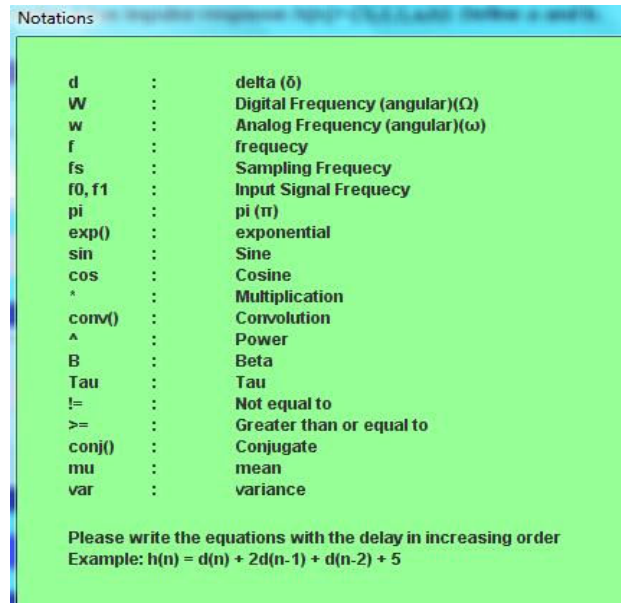


Figure 7. Notations Dialog.

While correcting the answers, the tolerance limit for the numerical answers is set to be ± 0.2 . (The answer is judged correct if it is in the range of ± 0.2 of the correct answer). For the plain text and equations, the sentence case, brackets and white spaces are ignored from the answers before performing the check. But even after considering many possibilities there can still be some cases when the automated correction does not work. For example, if the correct answer is $50*(x+y)$, but if the user enters $50x+50y$, the automated correction system fails. To tackle such cases, a feature has been developed which allows the instructor to enter more than one correct answer in the database. For example, now the instructor can type in both $50*(x+y)$ and $50x+50y$ in the space of correct answer, assuming that these two are the most common ways of writing this particular answer. The symbol '&' is used to separate multiple possibilities.

The other two types of questions are the multiple choice ones. Their answer could be either in the form of radio button or in the form of check box. The *question-answer* panel of figure 5 shows the radio button type of answers.

The purpose of this quiz is not only to test students' knowledge but to also give them a chance to learn from their mistakes. This motivated us to let students to have more than one attempt to answer the question. For the filling the blanks type of questions the number of maximum allowed attempts are fixed at three, while for multiple choice questions they are equal to number of options minus two if there are more than two options. Only one attempt is given if there are just two options. After the first attempt, the panel which contains the links to different learning environments (J-DSP demo, lecture notes, lecture video) is enabled. This helps student to understand the concept thoroughly and allows them to answer the question correctly in the

second attempt. It also helps in finding out whether the links to the other places were helpful or not.

Another quiz feature that verifies the usefulness of embedding the learning environments and also checks the students' learning curve is a concept of having a similar question. If the student can not answer a particular question correctly in the first attempt, the question following the current question will be very similar to the present question. This question would have been skipped, had the student answered the question correctly in first attempt. This feature helps in finding out whether the student has really learned the concept, after failing to answer the question correctly for the first time.

Links to other learning environments

As described earlier, the main motive of the quiz is to link multiple learning environments together. The links to these learning environments are activated as soon as the first attempt is made on a particular quiz question and are disabled again when the next question is presented. Figure 8(a) and Figure 8(b) show an illustration of the panel which contains these links, before and after the question is attempted. This panel can be found in Figure 5 also, where the entire quiz window is displayed.

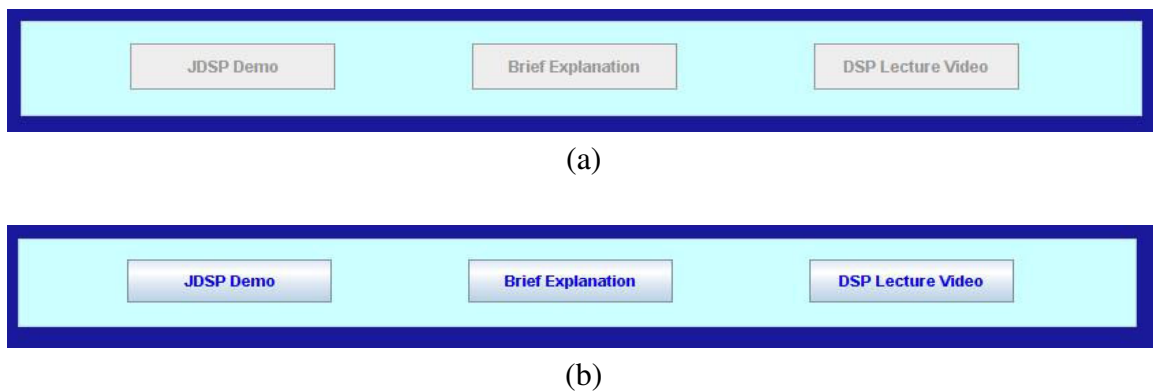


Figure 8. (a) Panel view before attempting a question. (b) Panel view after attempting a question.

The first link in the panel takes user to the animated J-DSP simulation demo. This J-DSP demo simulates a question and allows user to visually understand the concept. As soon as the JDSP Demo button is pressed, the relevant animation starts in the J-DSP simulation window #2. Figure 9 shows snapshots of the simulation area at the various time instants when the animation is in progress. This way, the quiz is directly connected to the J-DSP interface. Buttons are provided to change the speed of the animation or to skip the entire animation. The advantage of such a demo is that, it is not limited to just an animation where users can only view the construction of the block diagram or can view the results. In the background, this animation also constructs the actual block diagram in the simulation area. This allows the users to not only view but also to work on the constructed block diagram. This way they can find out the effect of changing the different parameters on the results and can get better insight into the theories.

The next link takes users to the relevant part of the DSP lecture notes. When users click the Brief Explanation button, a new tab is opened in a browser, where the slides of the lectures are displayed in the *pdf* format. This is a very important component as students generally refer to the lecture notes more often than anything and these notes also cover some theoretical material. This functionality lets them to have a look at the relevant part very quickly. There is also a comment section just besides the correct answer field (Figure 5). This section can be helpful in putting some additional notes which may not have been covered by the lecture slides. A brief explanation about the animated demo can be written here to guide students to view the results properly. Even some external web-links can be put in this section to give additional information about the topic in the question.

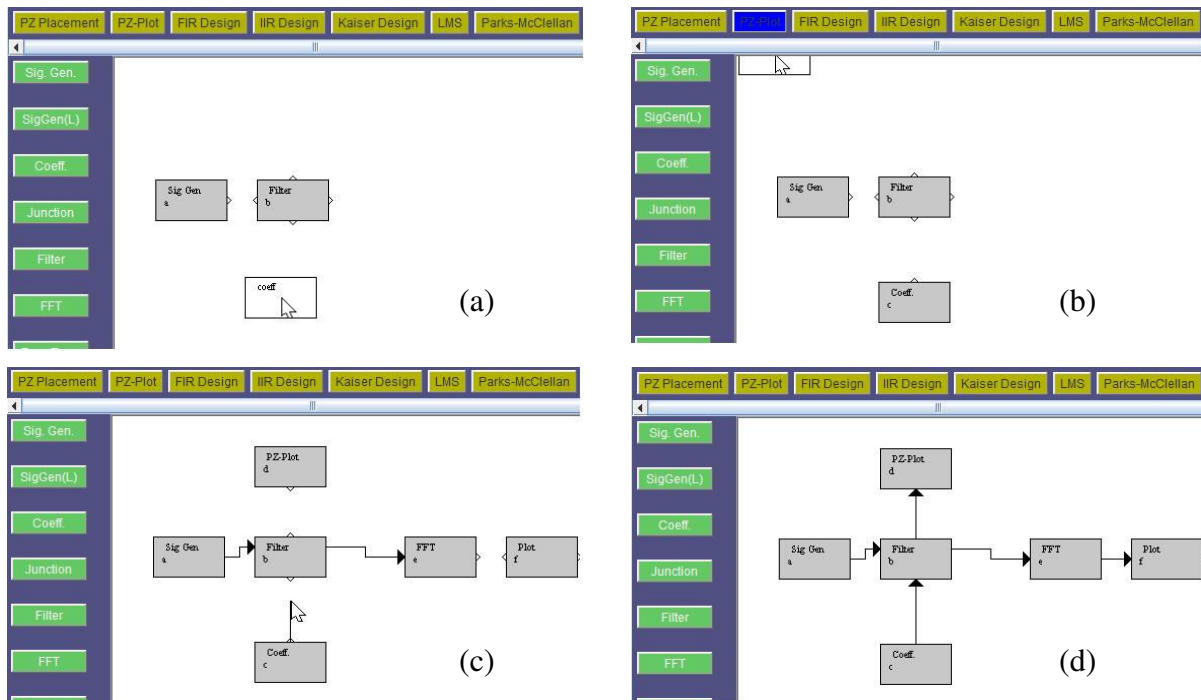


Figure 9. Animation of J-DSP simulation at various time instants.

The next and the final link, connects students to the relevant part of the lecture video. A snapshot of this is shown in Figure 10. This is one of the most helpful components as video lectures are always more interactive and they cover the theories along with the numerical examples. Students can refresh their concepts by listening to them again. This completes the entire interface of learning environments.

Editing the Quiz Chapters

The quiz edit utility simplifies the process of updating the database of the web-based quiz. Since the database can only be viewed or updated by the instructor, it requires administrative privileges to perform this task. The details of the privileges are covered in the next section. The conventional process of updating databases requires the instructor to log into the web server where the files containing the quiz questions are stored and download the files that need to be

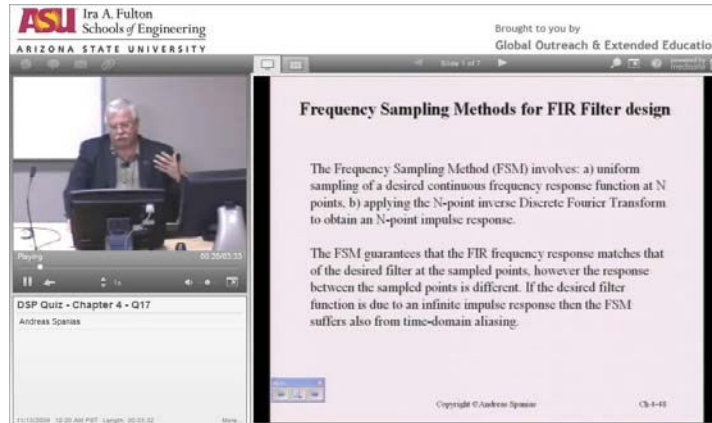


Figure 10. Screenshot of a lecture video.

updated. Then the modifications are performed by opening the files on the local machine and these updated files are again uploaded back to the server. The process becomes cumbersome when the required changes are minor and frequent. The interface proposed in this framework requires the instructor to just log into the J-DSP environment and open a quiz edit window from the menu items. This spreadsheet type of interface is very interactive and fairly simple to use as shown in Figure 11. It allows instructor to quickly view and update the database of the chapters that need to be modified. This feature is again embedded as a tabbed window into the J-DSP interface as shown in Figure 11.

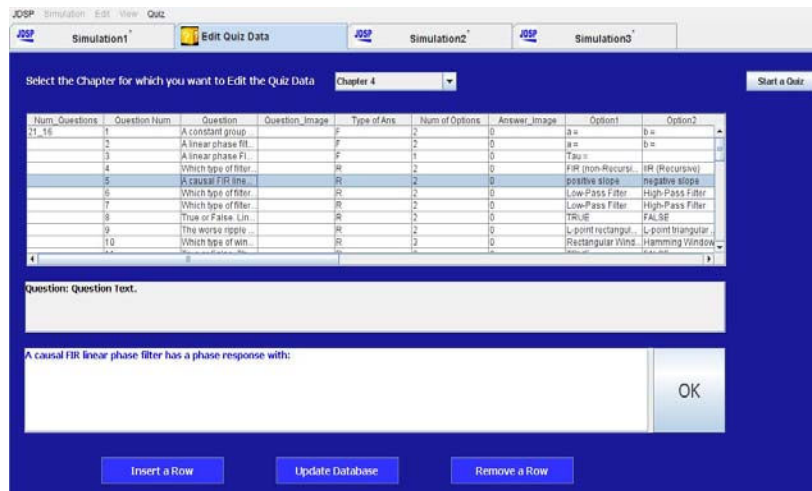


Figure 11. Quiz edit interface.

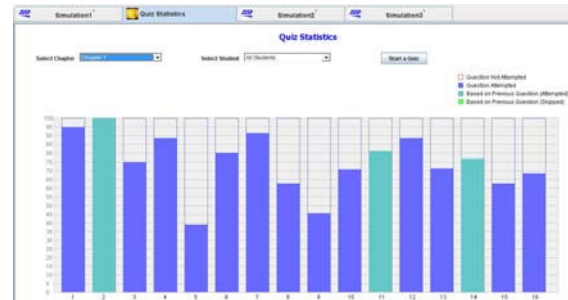
Quiz Statistics

This feature allows instructor to view the distribution of students' performance on the quiz questions. This utility is again available only for an administrator. Figure 12(a), (b), (c), and (d) show the four possible types of distributions available through this functionality. The first chart shows the performance of all the students on each chapter (Figure 12 (a)). Instructor can also zoom into a particular chapter to view the performance of all the students on each question of

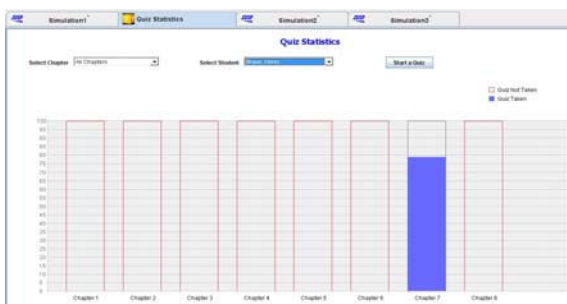
that chapter as shown in Figure 12 (b). The next distribution targets the performance of an individual student. Figure 12 (c) shows the performance of a particular student on each of the chapters. The last distribution shown in Figure 12 (d) is a chart of a performance of one student on a particular chapter. Undergraduate students of DSP class were asked to take this quiz. All the charts shown have been generated by collecting the data of their performance results.



(a) All students vs. all chapters.



(b) All students vs. a particular chapter.



(c) One student vs. all chapters.



(d) One student vs. one chapter.

Figure 12. Students' performance charts

Assessment Results

The aim of the proposed interface was to enhance the learning experience of the students by providing them with a complete integrated learning framework. Therefore the primary goal of this evaluation was to find out if there was any improvement in the understanding of various concepts after accessing different modules of learning environments. The assessment results are generated by asking students to answer a list of multiple choice questions about the usefulness of the proposed interface.

Two types of questions were asked. One set of questions was used to evaluate the usefulness of the links associated with the specific quiz questions. These types of questions asked students if the learning environments associated with a particular question were helpful. Figure 13 shows the responses to these questions. As shown, the assessment results are very encouraging. Almost 85% of the students agreed that they received help from all the three environments linked with the quiz questions.

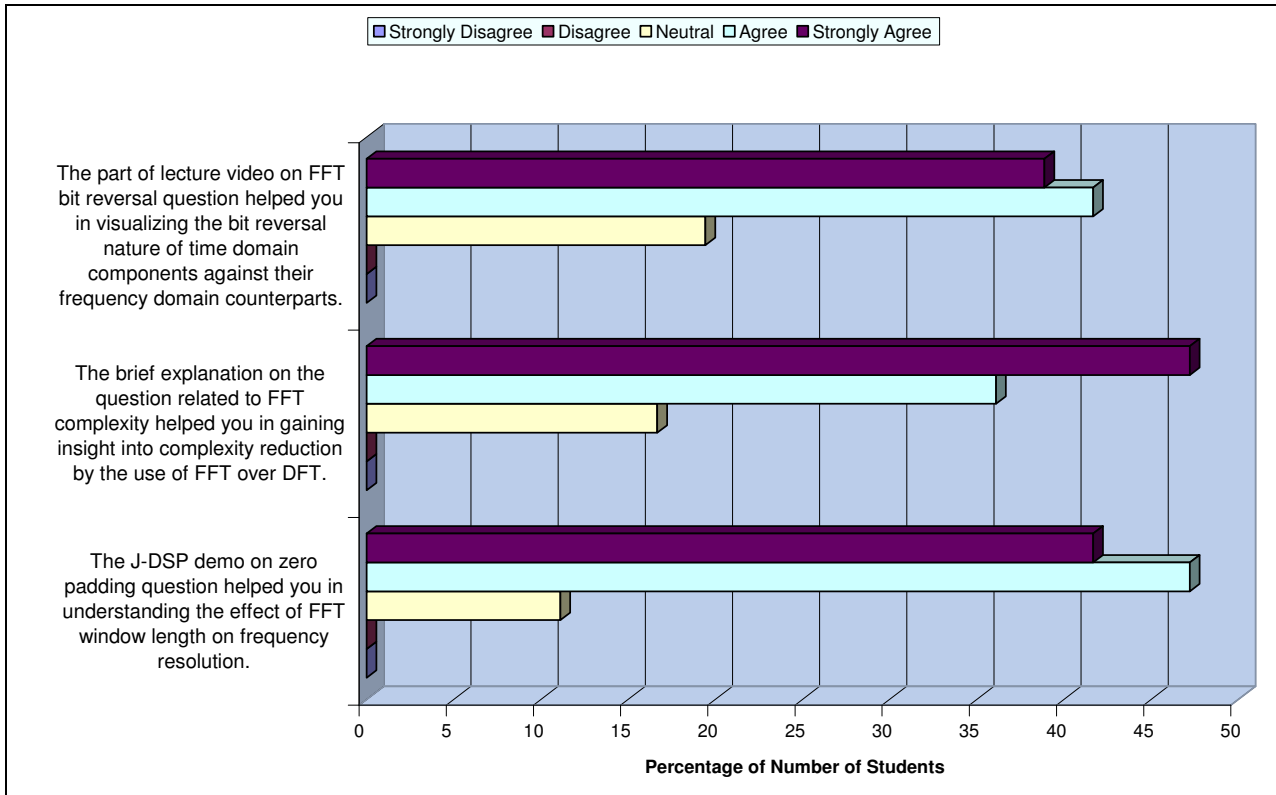


Figure 13. Students feedback on different learning environments.

The second type of questions was more general and asked students if they liked the current interface and its features. It also asked about evaluating their overall learning experience. The results are shown in Table I. Again the results are very promising and show that the students liked the features of the proposed interface. Almost 89% of the students agreed that the new framework provided them with an enhanced learning experience.

Table I. Statistics based on general evaluation of new interface.

| Question | Strongly Agree (%) | Agree (%) | Neutral (%) | Disagree (%) | Strongly Disagree (%) |
|--|--------------------|-----------|-------------|--------------|-----------------------|
| The simulation features in the new J-DSP interface are very useful. | 41.7 | 50.0 | 5.6 | 2.8 | 0.0 |
| The idea of embedding quiz/labs in the J-DSP simulation environment enhanced your learning experience. | 50.0 | 38.9 | 11.1 | 0.0 | 0.0 |
| The colors and the aesthetics (look and feel) of the new interface and quiz are pleasant. | 30.5 | 55.5 | 8.3 | 5.6 | 0.0 |

Conclusions

A new interactive J-DSP interface has been proposed in this paper. The main advantage of this interface is its integration with other learning environments. In addition to using the existing features of the current J-DSP, it includes the capabilities of other learning environments such as video, animations, etc. By integrating streaming lecture videos, slides, animated J-DSP simulation demos, additional notes and quiz statistics, this version of J-DSP becomes a complete online learning facility. The web-based quiz has a number of attractive features with the main focus on the improvement of students' learning. Interfaces have also been integrated to help instructors closely analyze the students' performance. The assessment results are also encouraging. The feedback from the students shows that they are finding this interface interactive and helpful for improving their learning.

Acknowledgements

This work is supported in part by NSF CCLI Phase 3 Grant # 0817596.

Bibliography

1. A. Clausen, A. Spanias, A. Xavier, and M. Tampi, "A Java signal analysis tool for signal processing experiments," in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, vol. 3, pp. 1849-1852, May 1998.
2. A. Spanias et al "Development of a web-based signal and speech processing laboratory for distance learning," ASEE Computers in Education Journal, pp. 21-26, vol. X, no.2, April-June 2000.
3. A. Spanias, F. Bizuneh, "Development of new functions and scripting capabilities in Java-DSP for easy creation and seamless integration of animated DSP simulations in Web courses", in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, May 2001.
4. R. Rangachar and A. Spanias, "A Simulation tool for introducing MPEG – Audio (MP3) concepts in a DSP course," in Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, May 2002, Orlando, Florida.
5. A. Spanias, K. Ahmed, A. Papandreou-Suppappola, and M. Zaman, "Assessment of the Java-Dsp (J-DSP) On-Line Laboratory Software," in Proceedings Of 33rd ASEE/IEEE FIE-03, Boulder, November 2003
6. A. Spanias, C. Panayiotou, T. Thrassyvoulou, and V. Atti, "Java-DSP interface with MATLAB and its use in engineering education," in Proceedings of IEEEASEE-2004 Conference, June 2004, Salt Lake City, Utah.
7. A. Spanias et al, "Teaching genomics and bioinformatics to undergraduates using J-DSP," in Proceedings of IEEE ASEE-2004 Conference, June 2004, Salt Lake City, Utah.
8. V. Atti, A. Spanias, C. Panayiotou, and Y. Song, "Teaching digital filter design techniques used in high-fidelity audio application," in Proceedings of IEEE ASEE-2004 Conference, June 2004, Salt Lake City, Utah.
9. A. Spanias, C. Panayiotou, and V. Atti, "Graphical design of frequency sampling filters for use in a signals and systems laboratory," in Proceedings of 34th ASEE/IEEE FIE-04, Oct. 2004, Savannah.
10. A. Spanias, R. Chilumula, C. Huang et al, "A Collaborative Project on Java-DSP involving Five Universities," in Proceedings of ASEE Annual Conference and Exposition, June 18-21, Chicago, 2006.
11. A. Spanias et al, "On-line laboratories for speech and image processing and for communication systems using J-DSP," in Proceedings Of 10th DSP workshop and 2nd Signal Processing Education Workshop, Gallaway Gardens, Pine Mountain, Georgia, 2002.
12. M. Yasin, L. J. Karam, and A. Spanias, "On-line laboratories for image and two-dimensional signal processing," in Proceedings of IEEE Frontiers in Education (FIE-2003), Nov. 2003, Boulder.
13. T. Thrassyvoulou, K. Tsakalis and A. Spanias, "J-DSP-C, A Control Systems Simulation Environment For Distance Learning: Labs And Assessment," in Proceedings Of 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, CO, November 5-8, 2003.
14. M. Zaman, A. Papandreou-Suppappola and A. Spanias, "Advanced concepts in time-frequency signal processing made simple", in Frontiers in Education Conference, November 2003.

15. A. Spanias, T. Thrassyvoulou, C. Panayiotou, and Y. Song, "Using J-DSP to introduce communications and multimedia technologies to high schools," in Proceedings of IEEE Frontiers in Education (FIE-2003), Nov. 2003, Boulder.
16. Y. Ko, T. Duman, and A. Spanias, "J-DSP for communications," 33rd ASEE/IEEE FIE Conference, Boulder, November 2003.
17. K. Ramamurthy, A. Spanias, L. Hinnov, and J. Ogg, "On the use of J-DSP in Earth systems," Proceedings of ASEE Annual Conference and Exposition, Pittsburgh, PA, June 2008.
18. A. Spanias, Digital Signal Processing; An Interactive Approach, 370 pages, Textbook with JAVA exercises, ISBN: 978-1-4243-2524-5, Lulu Press On-demand Publishers <http://www.lulu.com/content/2581497>, Morrisville, NC, Sept. 2007.