MATHEMATICAL ANALYSIS TOOL-KIT FOR e-ENGINEERING

Bhavin V. Mehta[1], and Shruti B. Mehta[2]

Abstract - *The authors have developed a few network-aware and heterogeneous components or Beans (most commonly used ones) and assembled them as a general-purpose web-based toolkit for numerical, statistical, signal and graphical analysis. The toolkit is developed using JavaBeans, in a modular or component form for e-Learning applications in engineering using platform neutral architecture. JavaBeans is implementation of Java but as component software or "Bean", which can be used over and over again by various applications on or off the web. Due to the effortless scalability and portability of Beans, ore components can be easily added and the toolkit can be expanded as newer reusable and distributive applications are identified and/or developed. These components and other software components (developed by other researchers) can also be assembled using visual application builder tools available from independent software developers. All the above programs or components are modules, which any one can use or call or include within their application by just passing the needed parameters, just like one would use or call a subroutine, the only difference being, it can be done on the web across several continents. The paper will present these numerical, graphical, signal and statistical analysis "beans" and demonstrate their reusability in web-based courses like Statics, Dynamics, Signals and Systems and Hearing Science.*

Keywords - JavaBeans, Distributed, Components, Numerical, Signal Processing, Statistical, Graphical, Web-based.

Introduction

Every university, industry and government organization is talking about developing web based applications, using text, multimedia, 3-D, interactive, virtual and what not. There are hundreds, if not, thousands of tools available for numerical, Signal and statistical analysis, ranging from $50 to $5,000. Faculty developing web-based engineering applications always needs numerical analysis modules, signal processing tools and graphical results display tools. They also have to integrate these tools into their research applications or on-line courses. It will be very convenient for the developers to be able to easily access these tools on the web and integrate them into their applications, whether it is for research or teaching [7]. Beans automatically inherit all the characteristics of any other Java program which include object-oriented, platform independent and relatively secure applications plus the additional feature of being reusable, reconfigurable and distributed provided by JavaBeans. JavaBean components can be called across the web or can share and pass data within several distributed applications. One can create distributed applications by running one chunk of a JavaBean component on one computer and another chunk of that application on another computer in some other part of the world. JavaBean components are optimized for distributed application across the Internet and thus can very easily communicate with CORBA (Common Object Request Broker Architecture) which is becoming an industry standard for developing distributed application or OBR's (Object Request Brokers).

[1] Associate Professor, Department of Mechanical Engineering, Ohio University, Athens, Ohio 45701
E-mail: mehta@ohiou.edu, Web: http://www.ent.ohiou.edu/~mehta

[2] Web Programmer, School of Electrical Engineering and Computer Science,
Center for Innovation in Technology for Learning, Ohio University, Athens, Ohio 45701
E-mail: mehtas@ohio.edu, web: http://www.ent.ohiou.edu/~smehta

JavaBean component can also be used by Microsoft's ActiveX component using the bridge. A bridge is a program, which translates ACTIVEX API calls into corresponding JAVABEAN API, calls in a transparent fashion. JavaBeans model is an extension to the Java language that enables one to create a special kind of Java program called Java component or 'bean'.

Design and Development

Initially the following components are being developed as part of the tool-kit (figure 1).

[1] Numerical Analysis
- Matrix manipulations
- Solution of Linear Algebraic equation using LU decomposition
- Solution of ordinary differential equations by RKF45 (Runge-Kutta-Felhberg technique)
- Solution of Stiff Differential Equations using the Gear Method
- Cubic Splines for data fitting and interpolation

[2] Statistical Analysis
- Measure of Central Tendency (Mean, Variance, Median)
- Probability Distribution (Binomial, Normal, Gaussian and Poisson)
- Analysis of Variance
- Regression Analysis
- Factor Analysis

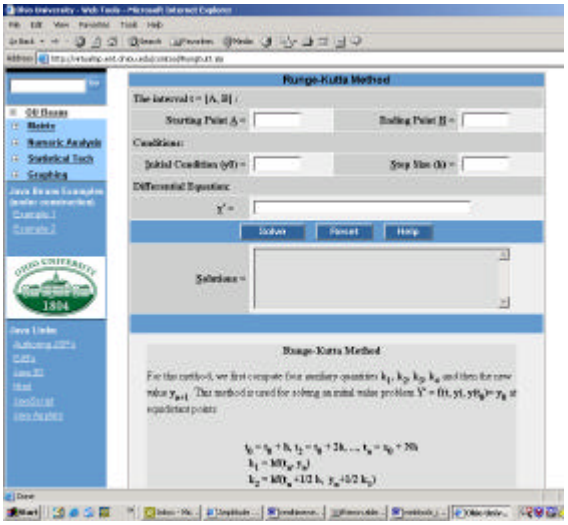
[3] Signal and Systems
- Amplitude Modulation
- Frequency Modulation

[4] Graphical
- XY plots
- Bar Charts
- Pie Charts
- Rotation about an arbitrary axis in 3D

[5] Unit conversion
- SI Units, English Units
- Special Units for other Engineering/Physics Applications

Figure 1: Front end of Web-based Tool-kit
http://www.ent.ohiou.edu/~comtool (click on OU Beans)

Beans will not only be able to interface with any other Java program or any Java supporting platform, but will also interface to relational databases and CORBA-complaint ORB's (via IIOP), OpenDoc components, Netscape plug-ins, and Microsoft's ActiveX components. Old legacy applications, which were developed for the mainframe can be interfaced to the beans, thus making it discipline and application neutral also. For example, if one looks at fundamental engineering courses or some physics or math courses, they all deal with developing the governing equation to solve a certain problem.



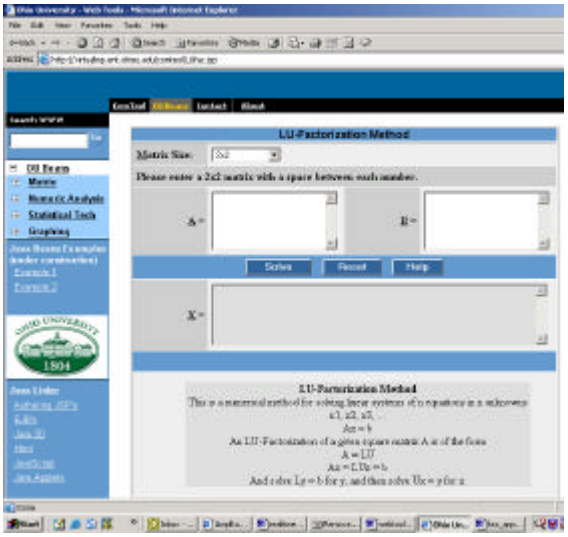


Figure 2a and 2b: LU Factorization and Runge-Kutta Beans

The differential equation or a system of equations has to be solved (in most cases) numerically [1]. For Example, the Runge-Kutta (RK4) component, as shown in figure 2a can be used by all those courses by just passing the differential equation to the RK4 bean/component. In most of those courses, the instructor does not have the time or scope to teach students how to solve the differential equation. In some cases the students might use a very simplified, inaccurate technique to solve the equation and get results, which are way off from the actual behavior. Since the components using JavaBeans are web based, the students can use it from their dorm rooms, or off campus. If the instructor wants to develop his/her course on the web, he will be able to use the RK4 or similar component transparently and would have more time to concentrate on the course itself, like developing the governing equation, instead of figuring out how to solve that equation (which is also needed). Once the results are generated using the numerical analysis tool, the graphical components can be called to visualize the results, again on the web. The student can develop the equation on a different computer, numerically solve the equation on another server (fast server for number crunching) and display the results graphically on a third computer (*illustrates the beauty of component based software*). The LU bean or the Guass-Siedel bean, as shown in figure 2b can be used to solve a system of equations is signal analysis and a complex signal can be plotted using the amplitude and frequency modulation beans. The figure 3a and 3b below shows the amplitude and frequency modulation plots.

The same graphical components to plot XY plots, bar charts or pie charts can be used by the Psychology department, or the Speech and Hearing department or the Education department or Biological sciences, after analyzing the data using the Statistical components to perform Regression Analysis, or Analysis of Variance. At the same time, the interpolation component can be used to fit the data, before plotting it. Again, all these tasks can be done just by calling the appropriate component in the tool-kit from their course related web pages.

The above two examples are just a few scenarios from a few hundreds, which demonstrates the re-usability of component-based software, the time savings for faculty developing web-based courses by being able to use the tool-kit, and thus a cost sav onent or r Java ap

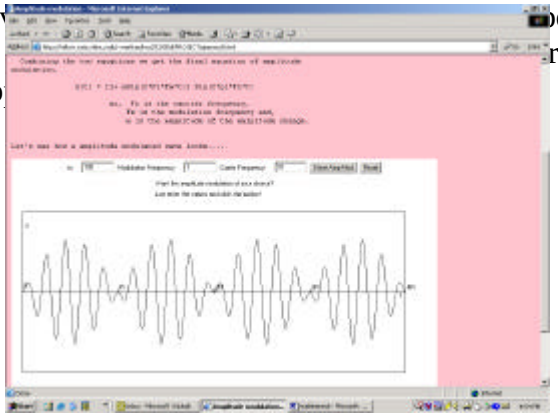


Figure 3a and 3b: Amplitude and Frequency Modulation Components

The user can customize the sequence of components called to solve a certain application from a menu as illustrated in example 3, and the data/results will be seamlessly transferred from one bean to another if the other bean requires it. For example, if the user calls the LU bean and then calls the Guass-Seidel bean only the input data is transferred but not the results. But if the user calls the RK4 bean and then the graphical bean then the results of the RK4 technique are transferred and plotted by the graphical bean.

Conclusion

The most widely used components have successfully been developed. The component/beans were used in the Statics, Dynamics [2, 3, 5], Signal and Systems and Hearing Science [4, 6] courses to solve differential equations, system of equations, plot graphs and signals, and convert units. The Beans can be used to analyze the system using different techniques and compare the results generated by those techniques as illustrated in the three examples on the web site. The use of the tool-kit is enormous due to its quick and efficient Internet delivery, scalability and reusability due to distributed architecture and user-friendly features.

Acknowledgement

The authors would like to thank Ohio Board of Regents (OBOR), Ohio University TIPs initiative, Center for Innovation in Technology for Learning (CITL) and Russ College of Engineering and Technology for funding this research and development.

References

1. B. V. Mehta and S. B. Mehta, Web-based Numerical Analysis Tools, 2nd *International Seminar on Numerical Analysis*, Batam, Indonesia, March 2001.
2. G. G. Kremer, and B. V. Mehta, Interactive Problem Solver for Mechanical Engineering on the World-Wide-Web, *Proceedings of the 2001 ASEE Annual Conference*, San Antonio, TX, June 2001.
3. B. V. Mehta, and G. G. Kremer, Dynamics of Design and Development of Introductory Web-based Engineering Mechanics Course, *Proceedings of WEBNET 2000*, San Antonio, TX, Nov. 2000.
4. S. B. Mehta, Pedagogical issues in teaching mathematical, signal processing and physics concepts to non majors, *Proceedings of WEBNET 2000*, San Antonio, Oct. 2000.
5. B. V. Mehta, and G. G. Kremer, Web-based Mechanics: Statics and Dynamics, *4th International Conference on Engineering Design*, Orlando, Florida, Aug. 2000.
6. S. B. Mehta, Development and Assessment of Interactive Web-based Problem Solvers in Reinforcing Math and Physics Concepts, *WEBNET 2001*, Orlando, Florida, October 2001.
7. A. Kovalchick, D. Mann, J. Bi, K. Sandell, S. B. Mehta, Providing Leadership in the design of Distributed Learning Environments, *Syllabus 99 Education Technology Conference*, Santa Clara, July 1999.