AC 2008-1844: WEB-BASED SIMULATION OF FLEXIBLE MANIPULATOR SYSTEMS

Abul Azad, Northern Illinois University

ABUL K M AZAD is an Associate Professor of Engineering Technology with the Technology Department of Northern Illinois University. He obtained Ph.D. (control engineering) from the University of Sheffield (UK) in 1994. He has worked at various academic and industrial establishments since graduation in 1987. His research interests include mechatronics, vibration control, real-time computer control of engineering systems, adaptive/intelligent control, and mobile robotics. In these areas, Dr. Azad has over 94 referred journal and conference papers, one edited book, and a book chapter. So far, he has attracted around $1.5M of research and development grants from national and international funding agencies. He is an editorial board member for a number of technical journals and international program committee member for conferences. Dr. Azad actively involved with other professional activities such as NSF proposal reviewer, European Commission Expert for 5th and 6th research framework, member for ISO standardization committees for robots in personal care and service robots, and an ABET program evaluator. He is a member of IEEE, ASEE, IET, and a senior member of ISA.

Ghayasuddin Mohammed, Northern Illinois University

GHYASUDDIN MOHAMMED was born in Karimnagar, India on February 15, 1984. He did his schooling in Warangal, Andhra Pradesh, India. In 2005, he received his Bachelor of Engineering Degree in Computer Science from Osmania University, Hyderabad, India. In 2005, he moved to US to pursue his MS (Computer Science) at Northern Illinois University and successfully graduated in Aug 2007. He is currently working as a Software Engineer – QA at Broadcom Corporation, San Diego, CA.

Mohammed Murtuza, Northern Illinois University

MOMAMMED MURTUZA was born in Hyderabad (southern part of India) on November 25th 1984. He came to USA in July 2006 in pursuit of Masters in Computer Science with the Northern Illinois University (NIU), where he enhanced his expertise and skills that acquired during Bachelors degree in Computer Science. In May 2006, he earned his Bachelors degree from Muffakham Jah College of Engineering and Technology, India that is affiliated with the Osmania University. During his graduate study with the NIU, he has an opportunity to work as a graduate assistant with the Technology Department of NIU. Presently he is working as a software engineer at ADP, Hoffman Estates, Illinois.

Muhammed Abedur Rahman, Northern Illinois University

MUHAMMED ABEDUR RAHMAN was born in Chittagong, Bangladesh on January 8th 1981. He did his schooling in Chittagong, Rajshahi, and Dhaka, three cities of Bangladesh. In 2004, Mr. Muhammed received a BS in computer science and engineering from the Rajshahi University of Engineering and Technology, Bangladesh. Since 2007 he is a student of Northern Illinois University, where he is pursuing his MS in computer science.

Study of Flexible Manipulator Systems: Web-based Virtual Facility for Modeling, Simulation, and Control

1 Introduction

In the engineering and technology arena, virtual facilities allow one to interact with a computer simulated environment that will provide experience with the operation and behavior of a real/imaginary engineering system with varying inputs and boundary conditions[1, 2]. A virtual facility can have two purposes: a) to facilitate the feasibility study of an engineering system while in the planning stage, and based on the study, designers can adjust the parameters during their final planning so the anticipated system can serve its purpose and b) to study the behavior and control of a complex engineering system for educational and research purposes without building the system. The development of these virtual facilities can involve expertise with complex mathematical solutions with tedious software developments and effective graphical user interface [3].

Considering the level of involvement, sometimes it is difficult for an individual institution to develop and maintain such facilities in terms of expertise and resources. To address this issue, with the advent of web and related technologies, it is possible to provide remote access to a developed virtual facility that will be housed at a central location. With this scenario, a group of experts can develop a virtual facility for an engineering system and make it available over the web from a remote server. Any change/modification of the developed facility needs to be done only within the server, which will allow a number of individuals or institutions to share their resources for a virtual facility. Otherwise it would have impossible to achieve this on individual basis. The lightweight flexible manipulators can be considered a complex system, and researchers/educators are trying to model and simulate these to develop effective controller designs [4, 5, 6]. Toward this effort, a web-based virtual facility for flexible manipulators would be useful. This paper addresses the development of a web-based virtual facility for modeling, simulating, and controlling flexible manipulator systems.

2. Benefits and Challenges of Flexible Manipulators

Lightweight flexible manipulator systems offer several advantages in contrast to their traditional rigid counterparts: faster system response, lower energy consumption, relatively smaller actuators, reduced nonlinearity due to elimination of gearing, less overall mass and, in general, less overall cost [7]. All these benefits attract attention for their use in various sectors of industry [8, 9]. However, apart from space programs (due to strict payload regulations for a spacecraft), their use has not yet that popular.

A main factor for achieving the advantages of flexible manipulators is the reduction of weight. However, this reduction of weight to a certain level gives way to a natural loss of stiffness, allowing the manipulator to vibrate due to elastic deformation during its motion and thus increasing the duration for completing a given task cycle (pick-position-place). A promising approach to compensate for this setback is to incorporate into these manipulators more sophisticated control algorithms with an involved actuation and sensing network. A basis for this approach is a manipulator model capable of reproducing the fundamental system dynamics in a given application and amenable for real-time computation. This discussion leads to the fact that effective controller designs for flexible manipulators are the key issue to make flexible manipulator systems attractive for the industries.

3. Theoretical Developments

As mentioned in the previous section, the full potential of flexible manipulator systems can't be explored until the vibration control issue can be resolved. To address this, along with controller designs, one needs to understand the behavior of flexible manipulators. With this scenario, the lead author has designed and developed a computer-based simulation facility for flexible manipulator systems. The facility uses Matlab and associated tool-boxes as the development package and was used to study the behavior of flexible manipulators along with controller designs. The facility includes a number of modeling and simulation schemes along with a few open-loop and closed-loop controller designs. This section will provide a general description of these schemes along with references to their theoretical basis. The lead author and a number of researchers from various countries around the world contributed to these theoretical developments. Contribution of these will be acknowledged through appropriate references.

3.1. Simulation and Modeling

Understanding the behavior of flexible manipulators and their interaction with actuators, sensors, and payload is a prerequisite for the development of an effective vibration control strategy and can be achieved by developing an appropriate mathematical model for the system[10]. Such a model can be constructed by solving partial differential equations (PDEs). The finite element (FE) and finite difference (FD) methods have also been utilized to describe the flexible behavior of manipulators [11]. The reported facility provides FE and FD simulation methods. In FE, the flexible manipulator is considered as an assemblage of a finite number of small elements. The elements are assumed interconnected at certain points known as nodes. For each finite element, the scalar kinetic and potential energy functions are formulated as functions of the generalized coordinates. A Lagrangian is formulated, and the dynamic model is obtained by applying Lagrange's equations. By reducing the element size, that is, by increasing the number of elements, the overall solution of the system equations can be made to converge to the exact solution as precisely as desired. The computational complexity and consequent software coding involved in the FE method is a major disadvantage of this technique [12]. The FD method is used to obtain an efficient numerical method of solving the PDE by developing a finite-dimensional simulation of a flexible manipulator system through a discretisation, both in time and space (distance) coordinates [13]. In this method, a set of equivalent difference equations defined by the central finite difference quotients of the FD method are obtained by discretising the PDE in equation with its associated boundary and initial conditions. The process involves dividing the manipulator into a number of sections and considering the deflection of each section at sample time points. The approaches have proven to be effective in simulation of such systems for test and verification of controller designs.

An alternative approach is to utilize intelligent techniques, such as genetic algorithm (GA) and neural network (NN) for modeling of flexible manipulator systems [14, 15]. GA algorithms form one of the prominent members of the broader class of evolutionary algorithms inspired by the mechanism of natural biological evolution, i.e., the principles of survival of the fittest [16]. The model operates on three basic steps: a) creation of an initial set of potential solutions known as population, b) evaluation of each solution and selection of the best ones, and c) genetic manipulation to create new population [17]. This is a closed loop process, and the average performance of individuals in a population is expected to increase, as good individuals are preserved and breed with one another and the less fit individuals die out. The GA is terminated when some criteria are satisfied, e.g., a certain number of generations completed or when a particular point in the search space is reached. NN possess various attractive features such as massive parallelism, distributed representation and computation, generalization ability, adaptability, and inherent contextual information processing [18]. Among the various types of NNs, the multi-layered perceptron (MLP) and radial basis function (RBF) are commonly utilized in identification and control of dynamic systems. An MLP-NN is capable of forming arbitrary decision boundaries and representing Boolean functions [19]. The network can be made up of any number of layers with a reasonable number of neurons in each layer, based on the nature of the application. A neuron performs two functions, namely, combining and activation. Different types of function such as threshold, piecewise linear, sigmoid, tansigmoid and Gaussian are used for activation.

3.2. Control

Vibration control techniques for flexible structures are generally classified into two categories: passive and active control [20]. Passive control utilizes the absorption property of matter and thus is realized by a fixed change in the physical parameters of the structure, for example adding viscoelastic material to increase the damping properties of the flexible manipulator [21, 22, 23]. Active control utilizes the principle of wave interference. This is realized by artificially generating anti-source(s) (actuator(s) to destructively interfere with the unwanted disturbances and thus result in reduction in the level of vibration. Active control of flexible manipulator systems can in general be divided into two categories: open-loop and closed-loop control. For the reported facility, the controller design involves both the classical open-loop and closed-loop controllers for flexible manipulator systems [24, 25].

Within the open-loop control low-pass filter band-stop filter and Gaussian shaped torque, inputs are considered. Initially, to identify the characteristic parameters of the system, the flexible manipulator is excited with a single-switch bang-bang torque input and its vibration behavior is monitored [24]. Then the filters are used for pre-processing the input so that no energy is fed into the system at the natural frequencies. Performances of the techniques are assessed in terms of level of vibration reduction at the natural frequencies, time response specifications, and robustness to natural frequency variation. These are accomplished by comparing the system response with the unshaped bang-bang input. The robustness of the control schemes is assessed with up to 30% tolerance in vibration frequencies. As the dynamic behavior and vibration of flexible manipulators is significantly affected by payload variations, the performance of the control strategies is also assessed with a flexible manipulator incorporating a payload.

A common strategy for closed-loop control of flexible manipulator systems involves the utilization of proportional and derivative (PD) feedback of collocated sensor signals, such as hub angle and hub velocity, and is known as joint-based collocated (JBC) control [25]. The JBC controllers are capable of reducing the vibration at the end-point of the manipulator as compared to a response with open-loop bang-bang input torque. However, for effective control of end-point vibration, it is necessary to use a further control loop accounting for flexural motion control of the system. A hybrid collocated and non-collocated control structure for control of a single-link flexible manipulator has been reported, where a PD configuration has been applied for control of the rigid body motion, and a proportional, integral, derivative (PID) control scheme with end-point displacement feedback has been used for vibration suppression of the manipulator [25]. The proposed collocated JBC and hybrid collocated and non-collocated control structure are capable of reducing the level of vibration at the end-point for fixed operating conditions.

Considering their performance limitations due to configuration change with payload variations, researchers also developed various adaptive and intelligent controller strategies [26, 27]. In the case of a flexible manipulator system, any change in payload mass will affect the system dynamics for which such a fixed controller will not be adequate. This problem can be addressed by making the developed controllers adaptive so that they can be adjusted according to changes in the system dynamics [28]. A self-tuning scheme is initially implemented using the pole assignment technique with JBC control. The hybrid collocated and non-collocated control scheme is then realized with an adaptive JBC position controller and an inverse end-point-model vibration controller. A recursive least squares algorithm is utilized to obtain an inverse model of the plant in parametric form. The problem of controller instability arising from the non-minimum phase characteristics exhibited in the plant model is resolved by reflecting the non-invertible zeros into the stability region. The performances of both schemes are investigated within a flexible manipulator simulation facility.

An alternative to the parametric approach described in the previous section, a neuro-inverse modeling approach can be adopted to realize the inverse plant model [29, 30]. This results in a neuron-inverse model control scheme. The neuro-controller thus obtained is used along with the adaptive JBC control to achieve both trajectory tracking and vibration suppression. It has been shown that an MLP neural network employing the backpropagation algorithm can approximate a wide range of non-linear functions to any desired accuracy [31]. The backpropagation learning algorithm is commonly used with MLP NNs [32].

4. Stand Alone Virtual Environment and Limitation

Based upon theoretical developments (as stated in the previous section), the lead author and other researchers developed a simulation environment called SCEFMAS. This was based on Matlab along with other associated toolboxes such as Guide, Simulink, signal processing, and control systems [10]. The major drawbacks of using the SCEFMAS are (a) users need to have Matlab software package within their system and (b) all the programs developed for the SCEFMAS need to be on the clients' computer. These issues limit the use of the facility, and any future versions of SCEFMAS need to be distributed physically to keep the package updated.

To address these problems, it was realized that if one can make the facility available over the web, this would allow the users to have access to the facility over the web with an anywhere-anytime arrangement and without any need of Matlab software package. At the same time, all the updates could be introduced to a single server without any time delay and administrative overhead. Recently, Matlab has introduced the Matlab Web Server package that allows one to access the Matlab programs and run them from a remote location over the web without having Matlab software running on clients' system. With this provision, the lead author introduced the flexible manipulator software facility (simulation, modeling, and control) over the web.

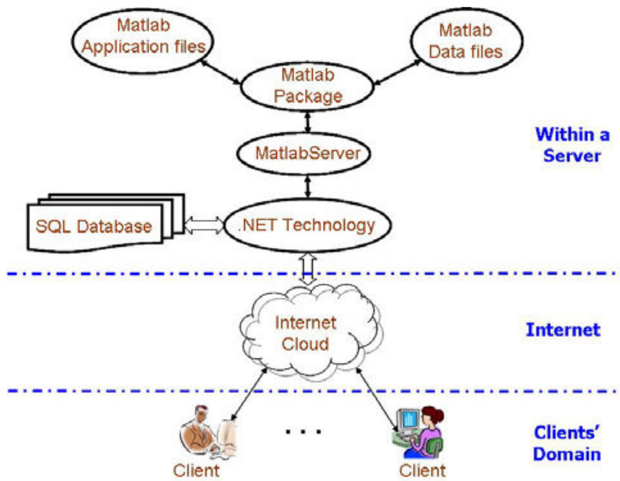

Figure 1: Shows major components and their integration within the facility.

5. Integration of the Facility

The facility is hosted within a server that incorporates a) Matlab application and data files; b) Matlab Web Server (MWS); and c) Web applications for access control, user account and password maintenance, administrative management, and facility utilization monitoring using ASP.NET technology and SQL database. Figure 1 shows a block diagram illustrating how all of the components are integrated together to make this a useful and effective system. All the Matlab application and data files required for simulation and control are stored within the server. The Matlab Web Server (MWS) provides a gateway between the ASP.NET applications and the Matlab applications. In addition to these, ASP.NET technologies along with SQL sever are used to develop web applications to provide graphical user interface for client, access control, administrative activities, and overall management of the system.

5.1 Matlab Web server

The Matlab Web Server (MWS) enables one to create Matlab applications within a server and use the capabilities of the web, allowing a client to send data to the server for computation and to display the results within a web browser at the client's end [33]. This arrangement depends upon TCP/IP networking for transmission of data between the client's system and MWS. In its simplest configuration, a web browser runs on a client's system, while Matlab, MWS, and web server daemon (httpd) run on the server side.

The MWS consists of a set of programs that facilitates communication between web applications and Matlab programs, which enable a client to access Matlab application programs over the web. It is a multithreaded TCP/IP server that invokes `matweb.m`, which in turn runs the Matlab application programs (M-file). The `matweb` is a TCP/IP client of MWS. The M-files were specified in a hidden field named `mlmfile` contained within the *html* document developed as web application. The MWS can be configured to listen on any legal TCP/IP port by editing the `matlabserver.conf` file. This program uses the common gateway interface to extract data from *html* documents and transfer it to *matlabserver*. All the client generated graphics must be located within the web server and subsequently passed to the client using appropriate web applications.

5.2 Matlab Applications

The facility includes a number of modeling and simulation provisions along with open- and closed-loop control strategies. In terms of modeling and simulation, there are four schemes- finite difference, finite element, neural networks, and genetic algorithm. Each of these implementations is developed through a number of Matlab files with one root file. The name of the root files for each of these simulation and modeling schemes are provided below:

Finite difference: *finite-diff-ws-fd*
Finite element: *model-ws-fe*
Neural network: *ws-nn*
Genetic algorithm: *ws-ga*

On the other hand, the controller implementation involves both the open- and closed-loop designs. Within the open-loop, there are a number of controller schemes: low pass filtered, band-stop filtered, and Gaussian shaped. For the closed-loop, the provided controller schemes are classical, adaptive, and intelligent control. Classical controllers are of joint-based and hybrid types, when the adaptive ones are adaptive JBC and adaptive hybrid and intelligent ones are adaptive neuron-inverse. There are two root files that implement all the controller designs:

Open-loop control: *ws-open*
Closed-loop control: *ws-closed*

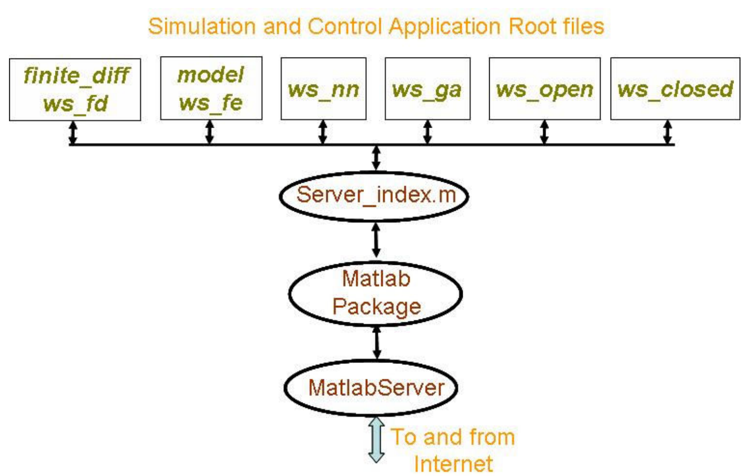

Figure 2: Interaction of Matlab applications with the Matlab web server.

The structure of the Matlab application files is presented through Figure 2. *Server_index.m* is the interface between the MWS and developed Matlab application files (root files). This file accepts input/output parameters requests from a client, calls specific root file for a specific request, and returns the results back to the client through Matlab web server.

6. Web Application

One of the major challenges in developing web application was to facilitate an interfacing between a client and a Matlab web server, while at the same time providing sufficient security, interactive graphical user interface, user tracking facility, and administrative capabilities for remote system management and data analysis.

6.1 Graphical User Interface/Application Pages

This section will provide details on the graphical user interfaces that are developed for the facility. An image of the home page is shown in Figure 3. The homepage provides a general description of the system along with links to the theoretical basis of various modeling, simulation, and controller designs. Following the login process, a client will be able to view the experiment page where they will be able to perform the experimental activities (Figure 4).

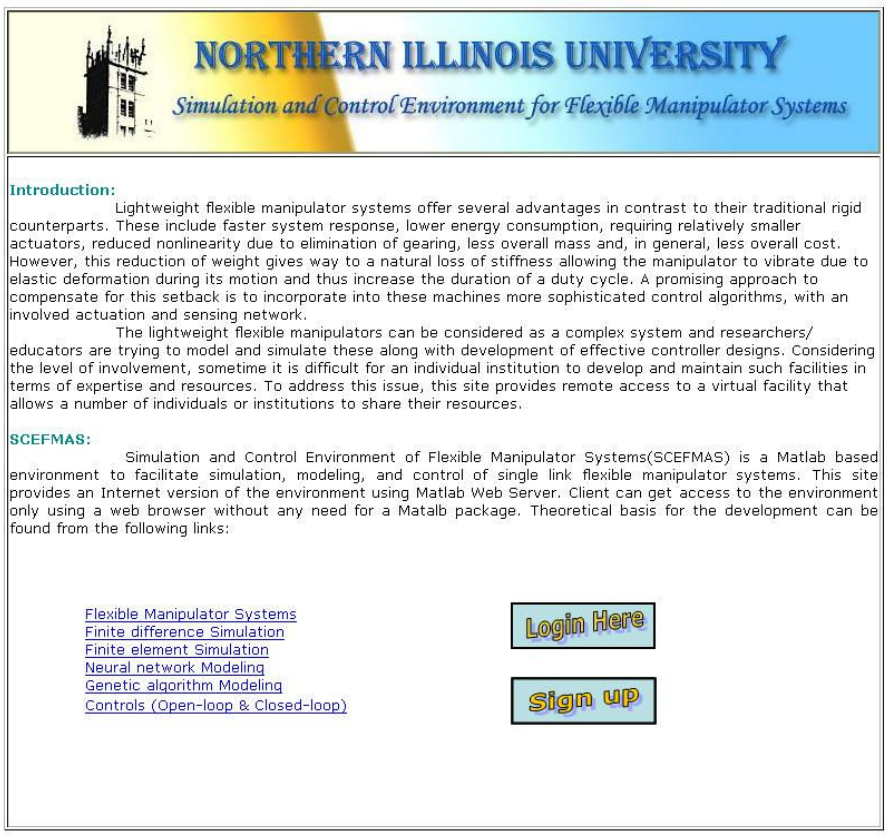

Figure 3: Home page of the developed facility.

6.1 Graphical User Interface/Application Pages

This section will provide details on the graphical user interfaces that are developed for the facility. An image of the home page is shown in Figure 3. The homepage provides a general description of the system along with links to the theoretical basis of various modeling, simulation, and controller designs. Following the login process, a client will be able to view the experiment page where they will be able to perform the experimental activities (Figure 4).

There are two steps to use the facility. The first step is to provide system specifications and simulation parameters. The system specifications include manipulator parameters along with material properties used for a desired manipulator. The simulation parameters include the duration of simulation and other simulation criterion. The second step is to select the input signal (excitation for the flexible manipulator system), desired algorithm, and required output graph.

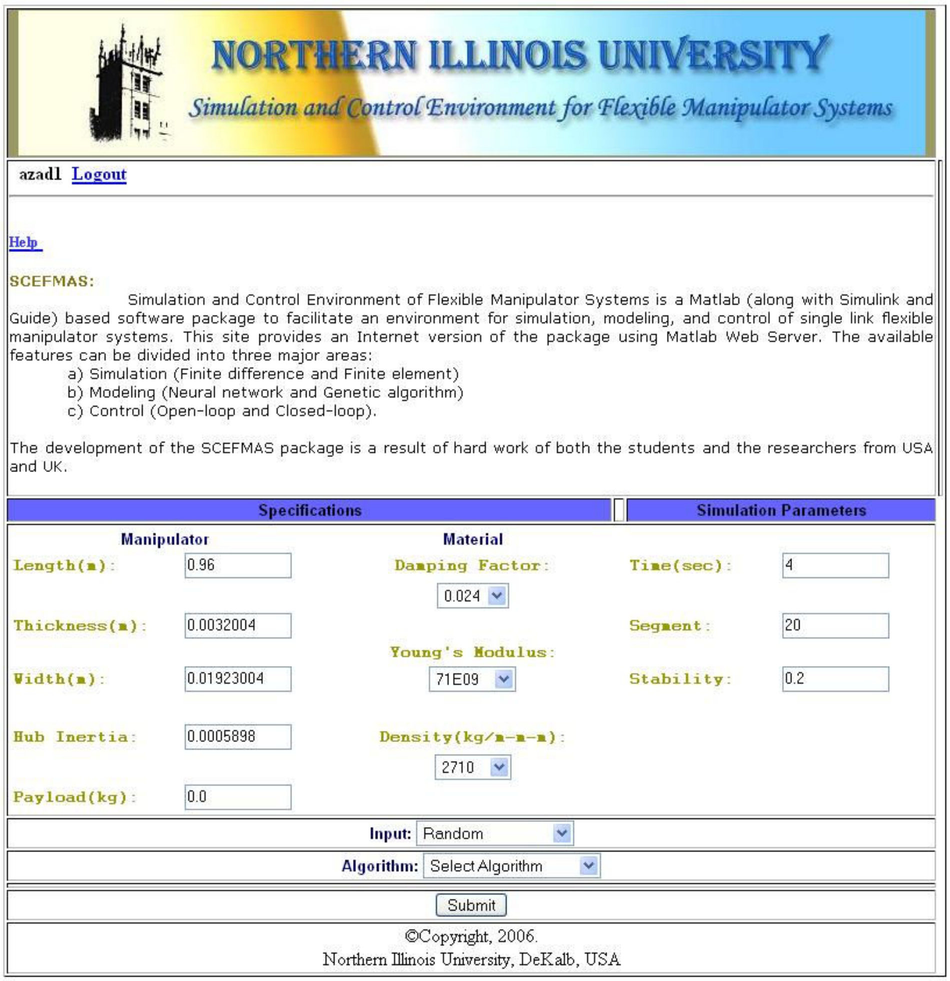

Figure 4: An image of the experiment page of facility.

There are three inputs types that can be used as the input torque to the system: bang-bang, random, and composite pseudo random binary sequence. Algorithm types are the available modeling and control schemes. There are a number of input and output graphs that can be available through this facility. This is to allow the users to study the behavior of the system and performance of the controller designs. Considering the limitation of MWS, one can view only one graph at a time.

For FD, FE, and controller designs, the available graphs are input torque, end-point displacement, end-point velocity, end-point acceleration, hub-angle, hub-velocity, hub-acceleration, and end-point residual. These are both in time and frequency domains.

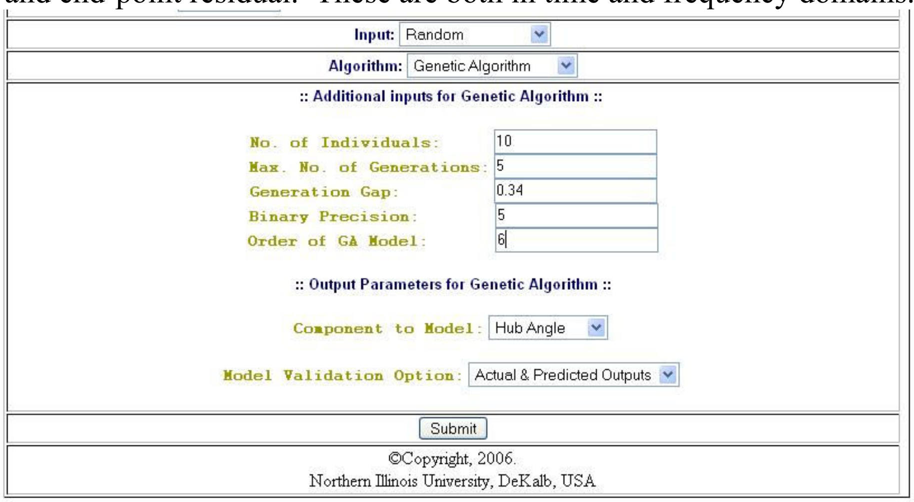

Figure 5: Image shows additional parameters for GA modeling.

While working with GA models, one needs to provide additional parameters. These are the number of individuals, maximum number of generations, generation gap, binary precision, and order of GA model. Users also need to choose one of the three models that are provided through this facility: hub angle, hub velocity, and hub acceleration. Models can be validated through observing the error for model predicted outputs. The available validation plots are the actual and predicted outputs, prediction error, sum of square error, and output spectral density. An image of the additional inputs for GA modeling is shown in Figure 5.

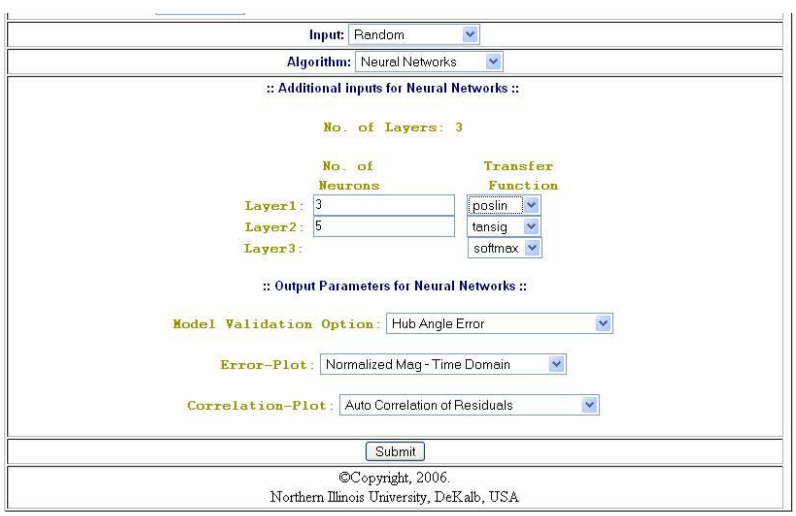

Figure 6: Image shows additional parameters for NN modeling.

NN modeling also requires additional input for modeling, such as the number of neurons in each layer and the transfer function of neurons in each layer. An image of the additional inputs for NN modeling along with the choice of validation for NN modeling is shown in Figure 6. Only three layers of the modeling structure are provided within this development. There are three types of models that can be done for NN modeling: hub angle, hub velocity, and end-point acceleration. Models can be validated through observing the error for the model predicted output and correlation tests. For the first method there are three graphs: time plot, normalized magnitude time domain, and normalized magnitude frequency domain. While for the second, the graphs are auto correlation residuals, cross correlation (CC) of input and residual, CC of square of the inputs and residuals, and CC of square of the inputs and square of the residuals.

6.2 *Client Access and Security*

The Matlab web-based tool is developed by using an Internet information services (IIS) server, which provides the services to the http requests coming through the web. This is a component provided within Windows 2000 and Windows XP Professional. The IIS makes it easier to share documents and information over the web. In the recent version of IIS, web-publishing, security, administration, and applications can work together to increase performance and reliability while lowering the cost of ownership and also improving the web application facility. The developed facility is provided with a password controlled security system. A SQL database holds all username corresponding passwords and some other details of all registered users. Figure 7 shows the relationship diagram for the developed database. For each user, data will be stored for login time, logout time, session ID, year, month, and day.

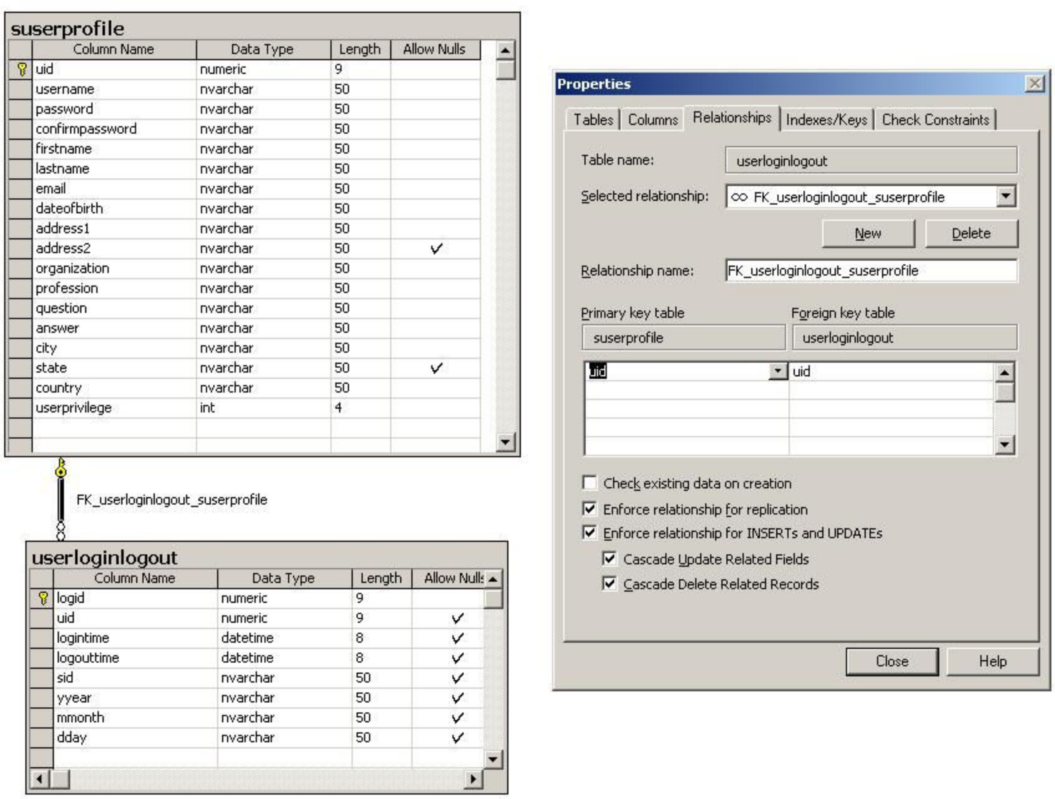

Figure 7: Relationship diagram for the developed database.

One has to register as a client by providing some details and creating a username and password. A client will have the ability to change the password and update his/her account details as necessary. There is a provision for extracting passwords and usernames if one forgets these. All these have been implemented using ASP.NET in CSharp. The ASP.NET has been chosen because of its simplicity and effectiveness, and the ASP.NET applications can run within the IIS with the facility's web page [34, 35]. The ASP.NET is also used to display the time and date along with maintaining the hit count for the website.

6.3 *Administrative Capabilities and Tracking*

To maintain the facility and to be able to monitor its use, the facility is equipped with administrative capabilities. These are override client abilities, facility use data tracking, generate system use statistics in terms of client's location, affiliation, profession, and access time. An image of the administrative user page is shown in Figure 8. Administrative capability allows one to sort the collected data in terms of username and country of origin, organization, and profession of the user. This has been implemented deploying filters for these fields using the database. This arrangement allows a system administrator to evaluate the user profile along with level of use of the facility.

UserName	FirstName	LastName	Organization	Profession	Country	Logintime	Logouttime
mqmurtuza	Mohammed	Mohammed	NIU	Student	USA	12/13/2007 11:46:11 PM	12/13/2007 11:46:45 PM
azad	Abul	Azad	NIU	Teaching	USA	12/13/2007 11:48:50 PM	12/13/2007 11:48:53 PM
mqmurtuza	Mohammed	Mohammed	NIU	Student	USA	12/14/2007 12:18:34 AM	12/14/2007 12:19:10 AM
mqmurtuza	Mohammed	Mohammed	NIU	Student	USA	12/14/2007 12:19:00 AM	12/14/2007 12:19:10 AM
azad	Abul	Azad	NIU	Teaching	USA	12/14/2007 12:19:25 AM	12/14/2007 12:19:30 AM
mqmurtuza	Mohammed	Mohammed	NIU	Student	USA	12/14/2007 12:29:20 AM	12/14/2007 12:29:24 AM
azad	Abul	Azad	NIU	Teaching	USA	12/14/2007 4:01:29 PM	12/14/2007 4:32:21 PM
azad	Abul	Azad	NIU	Teaching	USA	12/14/2007 4:02:01 PM	12/14/2007 4:32:21 PM
azad	Abul	Azad	NIU	Teaching	USA	12/14/2007 4:32:02 PM	12/14/2007 4:32:21 PM
mqmurtuza	Mohammed	Mohammed	NIU	Student	USA	12/14/2007 4:32:30 PM	12/14/2007 4:32:50 PM

1 2 3 4

Figure 8: Display of data within the administrative page.

Considering the academic use of this facility, the administrator page will allow the course administrator to use this information (in addition to other course data) for assessment and also to study the students' learning behavior using this facility. This will enable the administrator to assess the usefulness of the developed system and also provide adjustments to make the system more efficient and effective.

7. Conclusions

A web-based facility for simulation, modeling, and control of a flexible manipulator system has been presented. The main algorithm is implemented using Matlab and associated toolboxes, while the web deployment is facilitated by using MWS, ASP.NET, and SQL database. The facility includes a number of modeling and simulation schemes along with few open-loop and closed-loop controller designs. The simulation and modeling schemes include FD, FE, GA, and NN, while the control strategies involve classical and advanced controller designs with both collocated and non-collocated approaches. Classical designs incorporate various combinations of PD, PID for hub angle and end-point controllers, while the advanced controller designs present adaptive joint-based controller, adaptive hybrid controller, and adaptive neuro-inverse controller.

The web access to the facility allows multiple institutions/organization to share the facility with minimal effort and nominal cost. At the same time, any future developments can be implemented to the central server, which removes any delay in deployment.

Administrative level of access allows the system manager to override client abilities, facility use data tracking, generate system use statistics in terms of client's location, affiliation, profession, and access time. Administrative capability allows one to sort the collected data in terms of username and country of origin, organization, and profession of the user. This has been implemented by deploying filters for these fields within the database. This arrangement allows a system administrator to evaluate the user profile along with level of use of the facility.

This facility can serve as a valuable educational/research tool for understanding the behavior of flexible manipulator systems and development of various controller designs using model-based and artificial-intelligence based approaches. The facility can be used as a computer-aided teaching facility and also a test-bed for newly designed controllers for flexible manipulator systems.

Bibliography

1. Brooks Jr., F. P. (1999). What's Real About Virtual Reality?, *IEEE Computer Graphics And Applications*, **19**(6).
2. Burdea, G. and Coffet, P. (2003). *Virtual Reality Technology, Second Edition*. Wiley-IEEE Press.
3. Dwivedy, S. K. and Eberhard, P. (2006). Dynamic analysis of flexible manipulators, a literature review, *Mechanism and Machine Theory*, **41**(7), pp. 749-777.
4. Tokhi, M. O. and Azad, A.K.M., Control of flexible manipulator systems, *Proceedings of IMechE-I: Journal of Systems and Control Engineering*, 1996, **210**, pp. 113-130.
5. Tokhi, M. O. and Azad, A.K.M., (1996). Collocated and non-collocated feedback control of flexible manipulator systems, *Machine Vibration*, **5**, pp.170-178.

6. Shaheed, M. H. Poerwanto, H. and Tokhi, M. O. (2005). Adaptive Inverse and Neuro-inverse Dynamic Active Vibration Control of a Single-link Flexible Manipulator, *Proceedings of IMechE, Part-I: Journal of Systems and Control Engineering*, **219**, pp. 431-448.
7. Azad, A. K. M., (1994). *Analysis and design of control mechanisms for flexible manipulator systems*, PhD Thesis, University of Sheffield, Department of Automatic Control and Systems Engineering.
8. Piedboeuf, J.-C. and Moore, B., (2002). On the foreshortening effects of a rotating flexible beam using different modeling methods, *Mechanics of Structure and Machines,* **30**(1), pp. 83-102.
9. C. Lange1, J.-C. Piedboeuf1, M. Gu and J. Kövecses (2007). *Flexible space manipulators: Modeling, simulation, ground validation and space operation* (Book chapter), Editors: M. O. Tokhi and A. K. M. Azad, Flexible Manipulators- Modeling, Simulation, and Control, IET Press, London.
10. Tokhi, M. O., Azad, A. K. M., and Powrwanto, H. (1999). SCEFMAS: An environment for dynamic characterization and control of flexible robot manipulators, *International Journal of Engineering Education*, **15**(3), pp. 213-226.
11. Tokhi, M. O., Mohammed, Z. and Azad, A. K. M. (1997). Finite difference and finite element approaches to dynamic modeling of a flexible manipulator, *Proceedings of IMechE-I: Journal of Systems and Control Engineering,* **211**(I2), pp. 145-156.
12. Kourmoulis, P. K. (1990). *Parallel processing in the simulation and control of flexible beam structures*, PhD thesis, Department of Automatic Control and Systems Engineering, The University of Sheffield, UK.
13. Tokhi, M. O., and Azad, A. K. M. (1995). Real-time finite difference simulation of a single-link flexible manipulator system incorporating hub inertia and payload, *Proceedings of IMechE-I: Journal of Systems and Control Engineering*, **209**(I1), pp.21-33.
14. Shaheed, M. H. and Tokhi, M. O. (2002). Dynamic modeling of a single-link flexible manipulator: parametric and non-parametric approaches, *Robotica*, **20**, pp. 93-109.
15. Tokhi, M. O., Mohamed, Z. and Shaheed, M. H. (2000). Dynamic modeling of a flexible manipulator system incorporating payload: theory and experiments, *Journal of Low Frequency Noise, Vibration and Active Control*, **19**(4), pp. 209-229.
16. Holland, J. (1975). *Adaptation in natural and artificial systems*, University of Michigan Press, USA.
17. Chipperfield, A. J., Fleming, P. J., Pohlheim, H. and Fonseca, C. (1994). A genetic algorithm toolbox for MATLAB. *Proceedings of the International Conference on Systems Engineering*, Coventry, UK, 6-8 September.
18. Jain, A. K. and Mohiuddin K. M. (1996). Artificial Neural Networks: A Tutorial. *Computer: The IEEE Computer Society Magazine,* **29**(3)*,* pp. 31-44.
19. Minsky, M. and Papert, S. (1969). *Perceptrons: An introduction to computational geometry*, MIT Press, Cambridge.
20. Tokhi, M. O. and Veres, S. M. (2002). Active sound and vibration control - Theory and applications, Institution of Electrical Engineers, London.
21. Aubrun, J.-N. (1980). Theory of the structures by low-authority controllers, *Journal of Guidance and Control*, **3**, pp.441-451.
22. Choi, S. B., Gandhi, M. V. and Thompson, B. S. (1988). An experimental investigation of an articulating robotic manipulator with a graphite epoxy arm, *Journal of Robotic Systems*, **5**(1), pp.73-79.
23. Thompson, B. S. and Sung, C. K. (1986). A variational formulation for the dynamic viscoelastic finite element analysis of robotic manipulators constructed from composite materials, *ASME Journal of Mechanisms, Transmissions, and Automation Design*, **106**(2), pp. 183-190.
24. Tokhi, M. O., and Azad, A. K. M. (1995). Active vibration suppression of flexible manipulator systems - open-loop control methods, *International Journal of Active Control*, **1**(1), pp.15-43.
25. Tokhi, M. O., and Azad, A. K. M. (1995). Active vibration suppression of flexible manipulator systems - Closed-loop control methods, *International Journal of Active Control*, **1**(2), pp.79-107.
26. Menq, C.-H. and Chen, J.-S., (1998). Dynamic model and payload-adaptive control of a flexible manipulator. *Proceedings of the IEEE Conference on Robotics and Automation*, Philadelphia, USA, pp. 448-453.

27. Moallem, M., Patel, R.V. and Khorasani, K., (1997). An intelligent manifold approach to tip position tracking of flexible multi-link manipulators. *The IEEE Transaction on Robotics and Automation,* **13**, pp. 823-837.

28. Mohamed, Z., Martins, J., Tokhi, M. O., Sa Da Costa, J. and Botto, M. A. (2005). Vibration control of a very flexible manipulator system, *Control Engineering Practice*, **13**(3), pp.267-277

29. Tokhi, M. O. (2004). Intelligent methods for active noise and vibration control, *Archives of Acoustics*, **29**(2), pp.259-273.

30. Shaheed, M. H. Poerwanto, H. and Tokhi, M. O. (2005). Adaptive Inverse and Neuro-inverse Dynamic Active Vibration Control of a Single-link Flexible Manipulator, *Proceedings of IMechE, Part-I: Journal of Systems and Control Engineering*, **219**, pp. 431-448.

31. Blum, E. K. and Li, L. K. (1991). Approximation theory and feedforward theory. *Neural Networks*, **4**, pp. 511-515.

32. Hagan, M. T. and Menhaj, M. B. (1994). Training feedforward networks with Marquardt algorithm. The *IEEE transactions on Neural Networks*, **5**(6), pp. 989-993.

33. Matlab web server demos (2007). *http://hl1.uni-mb.si/matlab_index.htm*, Matlab Inc, (viewed on January 17th 2007).

34. Microsoft Windows 2000 Server documentation, *Microsoft Corporation*, *http://www.microsoft.com/ windows2000/ en/server/iis/default.htm*, (viewed on 11th of August 2004).

35. S. Walther, (1998). *Active Server Pages Unleashed*, Sams.net Publishing.