INTERNET-BASED
CONTROL-DESIGN ENVIRONMENT
AND ITS USE IN EDUCATION

Heřman Mann*, **Michal Ševčenko****

Computing and Information Centre*, Dept. of Computer Science**
Czech Technical University, Zikova Street 4, CZ-166 35 Prague 6,
Czech Republic, {`mann,sevcenko`}`@vc.cvut.cz`

Abstract: A control-design environment complementing the MATLAB control synthesis toolset with a toolbox called DYNAST for efficient modeling and 'virtual prototyping' has been developed. While MATLAB and Simulink require manual formulation of equations and block diagrams characterizing the controlled system, DYNAST allows for the system simulation, thorough verification and diagnostics without any such time consuming and error prone drudgery. DYNAST, sitting on the Czech Technical University server, can communicate with MATLAB and Simulink installed on client computers directly across the Internet. DYNAST is freely accessible in a Web-based, on-line and e-mail modes on `http://virtual.cvut.cz/dyn/`. This control-design environment has been exploited in a Web-based course on modeling and simulation of controlled multidisciplinary systems.

Keywords: modeling, simulation, design, education, Internet

1. INTRODUCTION

Most of the popular simulation toolsets like MAT-LAB admit model descriptions in the form of block diagrams or equations. These descriptions suit well to the abstract and idealized models used in control synthesis. Using them, however, for 'virtual prototyping' – i.e. for thorough control design verifications and realistic dynamic studies – is too laborious, cumbersome and error prone. Equations describing the system model as well as a block diagram representing the equations must be formed manually before the block diagram can be submitted to a computer. In addition, the block-diagram oriented simulators usually encounter numerical problems with causality, algebraic loops, changes of the equation order, etc.

At the same time, professional associations call for radical changes in the control engineering curriculum as well as for new innovative approaches to students training (see, for example, (IEEE Control Systems, 1999)). They criticize the existing courses namely for

- discouraging young people from engineering study by overemphasis on theory and mathematics at the expense of practical engineering issues
- presenting 'textbook' problems carefully engineered to fit ready-made idealized theoretical models without having the students to undertake the modeling by themselves
- using computers to carry out old exercises of an old curriculum without radical modification of the curriculum to incorporate computers in a way fully exploiting their power

While newcomers in large organizations can fill the gaps in their training and experience by learning from old-timers, those starting in smaller enterprises have to struggle on their own. Mostly, they are also disadvantaged by their insufficient access to a powerful computation platform and software.
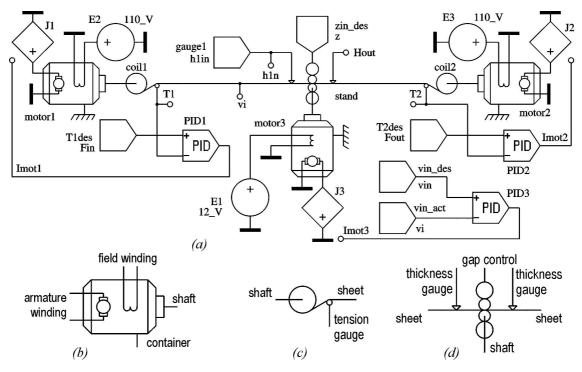
Fig. 1. (*a*) Rolling mill for metal strips. Multipole models of (*b*) DC motor, (*c*) strip coiler, (*d*) rolling stand.

Both in an engineering design and educational process, simulation software exploits computer capabilities efficiently if it not only solves the equations characterizing system models, but also formulates the equations. The contemporary engineering systems often require consideration of effects from several physical domains (like electrical, mechanical, fluid, acoustic or thermal) simultaneously. Thus, the equation formulation should be based on a multidisciplinary approach unifying the traditional approaches utilized in different engineering disciplines.

A software system called DYNAST has been developed with this aim about 15 years ago. Being now freely accessible across the Internet, it can be easily utilized by students and practicing engineers regardless of their geographical position. Together with MATLAB packages installed on their computers with which it can communicate across the Internet, it forms a very efficient control design environment that allows for avoiding any manual equation manipulation.

2. DYNAST SIMULATION SYSTEM

2.1 *Model descriptions*

Physical modeling is easy when using DYNAST, as it is based on the multipole modeling approach (Mann, 1995). The multipole models can be submitted to DYNAST in a graphical form isomorphic with the geometric configuration of the modeled real system. Therefore, a multipole model can be set up in a kit-like fashion based on mere inspection of the real system in the same way in which the system has been assembled from its real components.

Multipole models are represented graphically by multipole diagrams consisting of symbols for the individual multipoles. The symbols are interconnected by line segments representing energy interactions between the real components in a system. The variety of the multipole models can range from 'pure' two-pole elements like resistors, capacitors, inductors, dampers, inertors and springs up to sophisticated models of complex real components and subsystems.

Figure 1*a* shows a multipole model of a cold-rolling mill. Multipole models of the individual mill subsystems are shown in Figure 1*b* through *d*. The DC motors driving the coilers with metal-strip coils are controlled for a constant tension in the strip. The DC motor driving the milling rollers is controlled for a constant strip velocity. The gap between the rollers can be adjusted continuously by a hydraulic ram. The DC motor field windings are supplied by sources of voltage and the armature windings by controlled sources of current.

The multipoles in Figure 1*a* are combined with block diagrams modeling the controllers forming a dynamic diagram. Note the difference between the line segments interconnecting the blocks and the multipoles. The latter line segments represent in Figure 1*a* bidirectional energy transfers between components via idealized electrical conductors, shafts and the metal strip. Each of the line segments is associated with a pair of conjugate variables the product of which represents the transferred power. The segment interconnections respect physical laws governing the energy interactions.

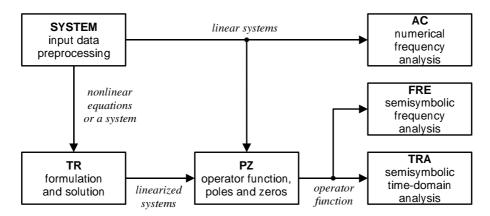The line segments interconnecting the blocks are associated with one mathematical variable only pro-

Fig. 2. DYNAST solver

pagating just in one direction only. The interactions respect algebraic rules, but ignore any physical laws. Obviously, while the block diagrams are nothing but graphical representations of equations, the multipole diagrams are mappings of real system geometric configurations onto their topological representations.

2.2 *Program structure*

The DYNAST Solver is composed of several sections sharing common data as shown in Figure 2. The section SYSTEM reads in the system-model description in the form of a set of algebro-differential equations, a block diagram, a multipole diagram, or in a form combining freely these approaches.

Nonlinear systems of equations can be solved, and nonlinear diagrams analyzed, in the TR section. This section computes system transient responses and system steady states, either static or periodic. In the former case, DYNAST automatically sets the time derivatives of all variables to zero. The static steady-states can be computed also for a system- or ambient-parameter sweeped through an interval. The transient responses can start either from initial conditions specified by the user, or from initial conditions corresponding to a static or periodic steady-state of the system. Fourier spectral analysis can be than applied to the periodic steady-state responses.

The TR section provides also automatic linearization of the analyzed nonlinear system. The resulting linearized system model can be subjected to small-excitation analysis in the vicinity of its user-specified or computed quiescent operating point. This analysis, yielding operator functions representing either system transfer functions or the transform of system initial-state responses, can be provided by the PZ section. The resulting operator functions are available in a semisymbolic form with the Laplace operator s as a symbol, and with the polynomial roots and coefficients as numbers. For such operator functions, DYNAST can than compute semisymbolic- and numeric-form time-domain characteristics using the TRA section, the FRE evaluates frequency characteristics nu-

merically. For linear systems, DYNAST provides also an option for their direct numerical frequency analysis in the AC section. This approach allows for the analysis of distributed-parameter dynamic systems (using hyperbolic and other frequency-dependent functions).

3. ACCESS TO DYNAST ACROSS THE INTERNET

3.1 *Web-based access*

DYNAST Solver providing the equation formulation and solution has been installed on a computer accessible via the Czech Technical University server. It can be accessed via `http://virtual.cvut.cz/dyn/` in a Web-based, on-line and e-mail modes. The Web-based mode function is illustrated by Figure3.

Setting up the multipole and block diagrams directly on the Web is enabled by the schematic editor DYN-CAD formed by a Java applet. One can use symbols of twopoles from different physical domains as well as symbols of transducers between physical domains and blocks. For more complex system components models stored in the DYNCAD libraries can be utilized, or users can define their own models and symbols. DYNCAD converts diagrams into the DYNAST input language and sends the data to DYNAST across the Internet. Users can open their free private accounts in DYNCAD and store there their simulation problems. DYNCAD is able to export the set-up diagrams into PostScript and to send them to the users by e-mail.

After the computational results are sent back to a client computer by the DYNAST server, there are two options for a graphical display of the computed responses on the client's screen. The plots generated on the DYNAST server in HTML are sent to the client computer as a Web-page. Or, the output data from DYNAST can be visualized on the client computer by means of a Java applet.
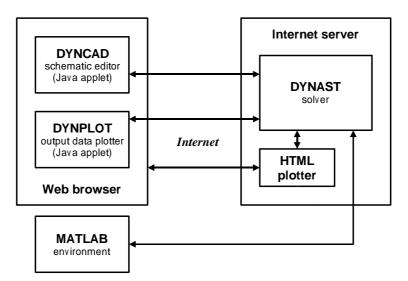
Fig. 3. DYNAST Web-based

3.2 *On-line access*

DYNAST can be utilized in an even more comfortable and user-friendly way using the on-line access mode illustrated in Figure 4. This mode requires, however, downloading and installing the DYNSHELL software package forming DYNAST user's environment for PC computers with MS Windows.

The DYNAST user's environment has been designed to provide a user-friendly support for a wide variety of tasks and suitable to users of different levels of qualification and experience. All operations are supported by a context sensitive help system. There is a built-in syntax analyzer continuously checking the submitted data. Dialog windows (wizards) allow for submitting data without knowledge of the input language. Multipole and block diagrams can be submitted in a graphical form using a built-in schematic editor. The same tool can be used to create submodel symbols. The diagrams can be imported also from the OrCAD schematic editor and converted to the DYNAST language.

Resulting data can be plotted in various arrangements and the plots can be exported in the Encapsulated PostScript format. The simulation results can be used for animation of 3D-space models formed using VRML- and Java-based visualization tools developed for the purpose. DYNSHELL can communicate with a server-based automated LaTeX documentation system generating reports for simulation experiments in the PostScript, PDF and HTML formats.

3.3 *E-mail access*

The e-mail access was designed for those with a limited access to the Internet. The users can send a file with the input data for DYNAST containing equations or the netlist of a dynamic diagram to the e-mail address: `DYNAST@icosym.cvut.cz` with the subject: `compute`. They then receive the results sent them back to their e-mail address automatically.

3.4 *Communication with MATLAB*

Using either DYNCAD or DYNSHELL, the plant model can be easily set up in a graphical form. DYNAST can be then used to simulate the plant and to validate its open-loop model. If the model is nonlinear, DYNAST is capable of linearizing it. DYNAST then can compute the required plant transfer-function poles and zeros, and export them to MATLAB in an M-file. After designing an analog control within the MATLAB environment, the DYNAST model of the plant to be controlled can be augmented by the designed control structure and thoroughly verified by DYNAST.

In the case of a digital control design, there is another option for the design verification. After designing the digital control in the MATLAB environment, the resulting control structure can be implemented in Simulink while the controlled plant model remains in DYNAST. Simulink installed on the client computer can then communicate with the remote DYNAST at each time step across the Internet exploiting the Simulink S-function.

In a similar way, also the cold-rolling mill shown in Figure 1*a* has been controlled. The hydraulic ram adjusting the roller gap can be controlled digitally across the Internet by a sophisticated adaptive controller implemented in Simulink and respecting the data coming across the Internet from the modeled gages and motors. During the simulation, the thickness of the incoming metal strip is varied randomly while a considerable transportation lag between the gage measurement and the rolling process is taken into account.
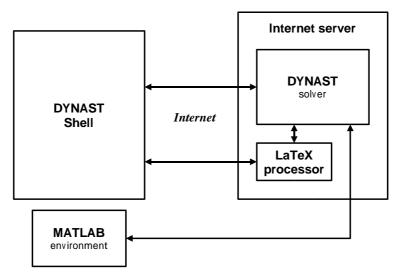
Fig. 4. DYNAST on-line

4. WEB-BASED COURSE

A Web-based course on unified modeling, simulation and analysis of multidisciplinary engineering systems (Mann, 1999) has been developed within an EU project (RichODL, 1999). The modeled systems may consist of components utilizing electrical, magnetic, electronic, mechanical, electro-mechanical, acoustic, electro-thermal, electro-hydraulic and other dynamic effects. The course is focused mainly on the multipole modeling approach giving the learners a systematic and efficient tool which does not distracts them from the modeling objectives by equation formulation, block diagram or bond graph construction.

The course is accompanied by a large collection of examples of various problems solved both in academia and industry to imitate knowledge sharing and informal learning typical for large organizations. The examples can be resolved and modified in an interactive way using DYNAST across the Internet. This gives the learners a hands-on opportunity to acquire the necessary skills in solving real-life problems. Thus the emphasis and style of the course differs from other existing courses by introducing learners to system dynamics and control-system modeling through simple yet practical examples. This stimulates the learners' interest in engineering and gives them a better 'feel' for the topic before exposing them to rigor math.

The main target groups of the course are

- regular face-to-face as well as distance education students of engineering schools at different levels of their study wishing to complement the traditional courses
- practicing engineers seeking education and training to upgrade their qualification in their job or to start a new carrier in the context of continuing education or lifelong learning
- teachers of engineering schools intending to innovate the courses they teach

5. CONCLUSIONS

The Internet based software tools presented in the paper form a core of the website of the *Virtual Action Group on Multidisciplinary System Simulation*, a part of the IEEE Control Systems Society Technical Committee on Computer Aided Control System Design `http://www-er.robotic.dlr.de/cacsd/`.

To demonstrate the efficiency of the presented control-design environment the simple modeling examples given in (Messner and Tilbury, 2000) were resolved and published on `http://virtual.cvut.cz/dyn/`.

6. REFERENCES

Future Directions in Control Education, special section in the *IEEE Control Systems*, Vol. 19, No. 5.

Mann, H. (1995). Multipole nd multiport approach to mixed energy-domain systems. Circuit model of energy-storing transducers. *Proc. 1995 IEEE Int. Symp. on Circuits and Systems*, Seattle, pp. 676-679 and 1271-1274.

Mann, H. (1999). Course on multipole modeling, simulation and analysis of multidisciplinary systems. `http://virtual.cvut.cz/course/`

RichODL (1999). Enriching ODL by knowledge sharing for collaborative computer-based modelling and simulation. *Socrates ODL Transnational Cooperation Project* No. 56057-CP-1-99. `http://virtual.cvut.cz/odl/`

Messner, B. and D. Tilbury (2000) Control Tutorials for MATLAB.

`http://www.engin.umich.edu/group/ctm/`