Interactive Tools for Education in Automatic Control

M. Johansson, M. Gäfvert and K. J. Åström

Automatic control covers a wide range of topics from mathematics to processes and computers. A good control engineer must master a wide variety of concepts, techniques and ideas. In addition, he or she must be able to apply them to real industrial problems. Typical tasks include mathematical modeling, analysis, simulation, design and implementation. The ability to solve these problems rests heavily on a number of concepts such as feedback, stability and robustness. Since dynamics is a key feature, it is necessary for a control engineer to have a very good understanding of dynamical systems, and that he or she be able to consider such systems from many different views. To instantiate a design, it is necessary to have a good understanding of and insight into trade-offs and compromises.

The need to provide students with both a strong theoretical base and engineering ability is a major challenge for education in automatic control. Theoretical issues, typically related to mathematical techniques, can be well taught in the ordinary classroom style. Engineering ability, on the other hand, requires insight and intuition, which are not so easy to convey. Previously, engineers acquired this through extensive laboratory work. A long time ago control engineers developed very good insight into dynamics by experimenting with analog computers. They could immediately see the consequences of changing parameters and system structure. During the past twenty years we have seen an amazing development of software for numerical computations and simulation. Although these systems are very powerful the man-machine interaction is still quite cumbersome. It often relies on command interaction, tedious iterations and a good knowledge of the software.

Because of the advances in software and hardware, it is now possible to design tools with much better man-machine interaction, with intuitive graphical user interfaces and a high degree of interactivity. Students can directly manipulate graphical representations of systems and get instant feedback on the effects. Different representations of system behavior can also be shown simultaneously. The tools allow students to explore different views of a system, manipulate views directly using the mouse, and immediately see the consequences on system behavior. The linear systems tool, for example, allows students to investigate over 10 systems per second on a standard PC with a 133Mhz Pentium processor. This is a drastic increase in the bandwidth of man-machine communication compared with conventional simulation.

To give the tools a high pedagogical value, we have implemented the learning environment as a collection of small modules. Each module is tailored to show a specific concept, illustrate a set of relations, or train a certain skill. In this way, each module has a clear focus, and the student has a good understanding of what issues she or he should try to master using a certain tool. Moreover, each tool is easy to learn and efficient to use. A high degree of interactivity makes the tools stimulating and quickly captures the interest of the user.

We have found the tools to be a nice complement to textbooks and laboratories. They are particularly useful for the purpose of developing skills and insight, but can also be used for conventional tasks such as analysis and design. A nice feature is their capability to abstract away minor issues and focus on essentials. The tools are available at any time, and allow students to work at home, at their own pace. In this paper we illustrate their use in the typical tasks of modeling, analysis, and design for a first course in automatic control. A companion paper [19] describes similar tools developed for a course in computer controlled systems.

Many tools for control education have been developed over the years. Much of the work on computer-aided control engineering originated from educational tools, see for instance [1], [2], [11], [15] and [17]. Many good ideas were developed by Professor Schaufelberger in a very ambitious project on computers in education at ETH, see [12]. Packages such as Matlab and MatrixX, with their toolboxes, have had a strong influence. Some of the toolboxes in Matlab have demos with significant educational value. The interactive control design module in MatrixX [5] is an excellent tool for developing a feel for dynamics and control. A nice set of computer-based controls tutorials that use the Web for distribution and instruction is described in [16]. There is also interesting software based on symbolic computations, such as the toolboxes in Mathematica (see [10]). The systems described in [6], [5] and [17] are closest to our work.

This work has been partially supported by the Swedish Research Council for Engineering Science under contract 95-759, and by grants on Computers in Education from Lund University. *The authors are with the Department of Automatic Control, Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden ({mikaelj, magnus}@control.lth.se).* *The article is based on papers presented at the IEEE CCA'96 and IEEE CDC'96.*

Experiments with educational tools have been carried out for a long time at Lund Institute of Technology. In the beginning we used special purpose analog equipment and later highly specialized software. Early implementations of interactive learning tools, much in the spirit of the software described in this paper, are described in [4] and [8]. The initial inspiration came from the use of spreadsheets, which revolutionized the business community. At this time, the capacity of computers was much more limited, and the implementation of interactive tools required a substantial effort. This restricted their educational use. The computer situation has now changed drastically. Good computer facilities are available and many students have computers at home, often with good Internet access. There has also been significant progress in software tools. It is now possible to implement interactive learning tools in standard software with reasonable effort. Experiments with Matlab~4 implementations are given in [9], [13], [14] and [18]. The textbook [3] is provided with a collection of tools of this type, see [19]. The programs described in this paper have been implemented in Matlab-5, and a toolbox with several additional tools is available for distribution [7].

Modeling of Dynamical Systems

Models and abstractions are central themes in control engineering. It is important that students understand how process behavior can be captured by mathematical models. They should be able to derive models, and to analyze their static and dynamic properties. Since so much of control theory is based on linear models, it is essential that students be familiar with linearization. They should know about the validity of linear models, and they should be aware of the limitations of linear analysis. To illustrate how interactive learning tools can enhance the understanding of these issues, we will consider a simple tank with a controlled inflow and an outflow that follows a square root law.

In this case, it may be desirable to consider several views of the system: a graphical illustration of the tank, a static characteristic, and the dynamic model of the system. Normally, we would represent these entities with drawings, graphs and equations which the student is requested to develop by pen and paper work. The physical system, for example, can be represented with a simple picture, the static input-output relation with a graph, and the dynamic model with a differential equation or perhaps also transient and frequency response.

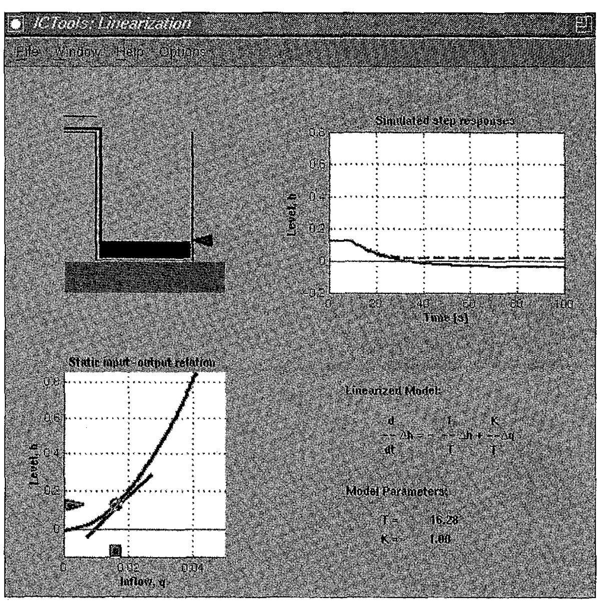


Fig. 1. Applying the suggested paradigm to a simple level control system.

The tank dynamics is given by

$$A\frac{d}{dt}h(t) = q_{in}(t) - q_{out}(t),$$
$$q_{out}(t) = a\sqrt{2gh(t)} \tag{1}$$

where A is the cross-sectional area of the tank, h is the height of the water in the tank, a is the area of the outflow tube, and g is the acceleration due to gravity. The steady state input-output relationship is given by

$$q_{in} = a\sqrt{2gh}. \tag{2}$$

This relationship is strongly nonlinear, particularly for small h. Linearization around an equilibrium $q_{in} = q_{out} = q_0 = a\sqrt{2gh_0}$ gives the model

$$\frac{d}{dt}\Delta h = -\frac{q_0}{2Ah_0}\Delta h + \frac{1}{A}\Delta q_{in}, \tag{3}$$

which can be characterized by the time constant $T = 2Ah_0 / q_0$ and the gain $K = T / A$. Both the time constant and gain vary significantly over the operating range. To develop a good intuition it is necessary to repeat calculations in order to see what happens when the characteristics of the system are changed.

In Figure 1, the system is represented by an interactive learning tool with four objects: a schematic diagram, a graph of the static relation (2) between inflow and tank level, step responses, and equations and time constants. The liquid level can be manipulated directly by dragging it, either in the schematic diagram or in the static input-output relationship. All other representations will immediately reflect the changes. In the static input-output graph, we also show the linearized relationship. The range of validity of the linearization can be judged from this representation. In the step response object, we show step responses for the linear (solid) and nonlinear (dashed) models. This is one way to visualize the validity of linearization. The magnitudes of the step responses can be manipulated directly in the diagrams.

By manipulating the objects in the tool directly, it is easy to obtain an intuitive understanding of the system. It is also possible to start a dialog with questions of the following type: How do the gain and the time constant vary with the level? How do the nonlinear and linearized models compare? What are the ranges of validity of the linearized model? For example, in the step response in Figure 1 we can observe that the linearized model (the solid line) predicts a negative level upon emptying the tank. This can be explained immediately by inspection of the curve representing the static input-output relationship.

Integrated Process and Control Design

The interplay between process design and control design is a key issue in control engineering. It is important for a process designer to be able to assess the consequences of process changes on process dynamics. If this can be done properly, it may be possible to avoid designs that will give rise to difficult control problems. The relationships between process design and process dynamics can be illustrated conveniently using a slight modification of the tank system in Figure 1. Assume, for example, that we have a tank which is not cylindrical. The properties of the system change drastically with changes in the tank level. Equations (1), (2), and (3) still hold, but A now depends on the liquid level. By allowing the student to change the cross section of the tank interactively, it is possible to explore the influence of the tank geometry on the static and dynamic properties of the system. Interesting questions to explore are: How does the tank geometry influence the control problem? How should the tank be changed in order to have a static relation that is close to linear over a given operating range?

Analysis of Dynamical Systems

When a model has been obtained, it is natural to investigate its properties. Analysis of dynamical systems in open- and closed-loops is a core topic in automatic control. Many powerful concepts and ideas have been developed to describe dynamical systems. Even for linear systems, there is a collection of tools that is somewhat bewildering to the student when first confronted with them. To develop good insight and intuition about the different representations and their interrelations constitutes a particular difficulty. Interactive learning tools can be very useful to enhance the understanding of different system representations.

Consider, for example, a linear time invariant dynamical system. Such a system can be represented by equations, time responses, or frequency responses. Equations can be given in terms of state space or transfer functions; time responses can be shown as impulse- and step responses. There are many useful representations of frequency responses, including the Bode, Nyquist and Nichols diagrams. The different representations are useful for different purposes. The different representations and their interrelations can be conveniently illustrated using a computer tool. A very good intuition for system descriptions can be developed using a tool that shows several different representations and allows for direct manipulation of the system.

Representations of Linear Systems

Figure 2 shows a tool that illustrates several views of a linear time-invariant system of arbitrary order. The main object of the tool is the pole/zero diagram. In addition, there are two objects that can show other representations of the system. They are selected via the options menu, which contains the choices: Nyquist curve, Bode magnitude plot, step response, and Bode phase plot. In Figure 2, we have chosen to display the the Nyquist curve and the step response. The pole/zero diagram is naturally manipulated by dragging the poles and zeros; the gain is represented by a slider. This figure illustrates the use of the tool for a linear system with two poles and one zero. It is, of course, not possible to demonstrate the power of a dynamic tool on paper. To give some flavor of the rubber-band effects that come from direct interaction and immediate plot updates, the illustrations show three pole positions. This shows the response of the tool when the poles are dragged from the lighter pole locations to the darker.

Many useful things can be learned from the tool shown in Figure 2. Natural questions to ask the students are: What happens when the pole locations are changed radially? What happens when the zero is changed? In particular, what happens when the zero is in the right half plane? What is the effect of gain changes? A very interesting dialog can be initiated starting with questions of this type. A natural followup is to have the students explain their observations based on their formal knowledge. In this way, the tools can be used to motivate the learning of the necessary mathematical manipulations.

In a pole/zero diagram, it is natural to manipulate the poles and zeros directly using the mouse. There are, however, two different representations of the transfer function that are of interest. One possibility is

$$G(s) = k\frac{(s-z_1)(s-z_2)\cdots}{(s-p_1)(s-p_2)\cdots} \tag{4}$$

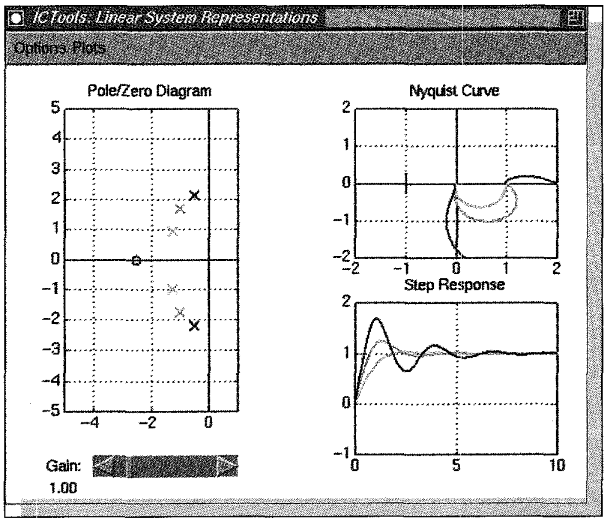


Fig. 2. Simple tool for analysis of a linear time invariant system.

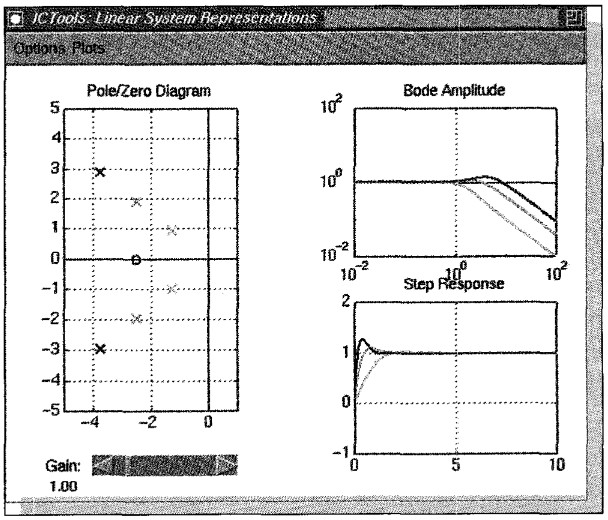


Fig. 3. Illustrating the relation between rise time and bandwidth when the poles are moved radially.

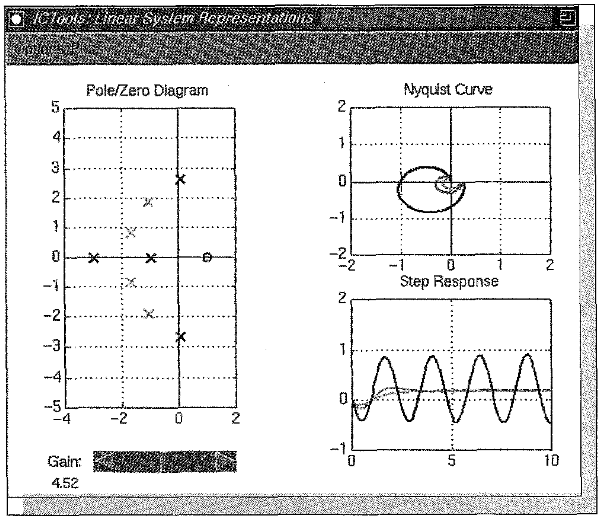


Fig. 4. Multiple views that illustrate stability and instability: singularities for open and closed systems, Nyquist curve and closed loop step response.

where k represents the high frequency gain. For many purposes, it is much more natural to manipulate the low frequency gain. This corresponds to the following form of the transfer function.

$$G(s) = k\frac{(s/z_1 - 1)(s/z_2 - 1)\cdots}{(s/p_1 - 1)(s/p_2 - 1)\cdots} \tag{5}$$

In Figure 2, we have chosen the representation (5) where the low frequency gain is invariant when poles and zeros are changed.

The tool can be reconfigured into many different variations by changing the additional plots from the options menu. In Figure 3, we show the gain curve instead of the Nyquist diagram. This view makes the tool well suited to illustrate the relationships between frequency domain and time domain. In this figure, we have shown what happens when the complex poles are dragged radially. Making the poles faster, moving the away from the origin, increases the bandwidth and decreases the settling time. The influence of the zero on the overshoot can also be well explored using this tool.

Open- and Closed-Loop Systems

The relationships between open- and closed-loop systems is fundamental for automatic control. It is, for example, very interesting to assess the changes in a closed-loop system due to changes in the process or the controller. These effects can be conveniently illustrated by the tool.

Figure 4 illustrates the case of a simple closed-loop system with a proportional controller and a process with the transfer function.

$$G(s) = \frac{1-s}{(s+1)(s+3)}$$

In closed-loop operation, the tool also shows the appropriate closed-loop properties. Color is used to separate the properties of open- and closed-loop systems. In Figure 4, the system is represented by three objects: a singularity diagram with poles and zeros of the open- and closed-loop systems; a Nyquist curve for the loop transfer function; and the step response of the closed-loop system.

By direct manipulation of the diagram, it is possible to obtain good insight into complicated relationships that are very tedious to obtain in any other way. Assume, for example, that we want to know how the properties of the closed-loop system are influenced by changes in the controller gain. Dragging the slider below the pole/zero diagram gives a clear illustration of the idea of the root locus. It is very informative to see how the different representations of the closed-loop system change with the gain. The closed-loop poles move towards the right half plane when controller gain is increased. The Nyquist curve is expanded radially and the step response becomes more oscillatory and even unstable. The closed loop poles cross the imaginary axis when the Nyquist curve intersects the critical point and the step response has a pure sinusoidal component. Students can be asked to determine the critical value of the gain and compare it with results obtained in a laboratory experiment.

Interesting experiments also include setting the gain so that the system is at the stability boundary and exploring the effects of changing open-loop poles and zeros. A typical question is: Will stability be improved when the zero is moved further into the right half plane? It is also interesting to see the consequences of introducing extra poles and zeros.

The step responses shown in Figure 4 have a steady-state offset. This can be removed by introducing a pole of the loop transfer function at the origin. It is interesting to investigate the consequences of this.

The idea of robustness of feedback can be illustrated by dragging the open-loop poles and zeros while observing the corresponding changes in the closed-loop system. Interesting questions are to explore which poles or zeros have the largest effect on the closed-loop system.

Design Limitations

When designing a control system, it is essential to have a good grasp of fundamental trade-offs and limitations. We have made several attempts to provide tools for this. One example is described in this section.

Consider the closed-loop system whose block diagram is shown in Figure 5.

The system has three inputs, r, d and n, and three interesting output signals, u, y and $e = r - y$. If P is the transfer function of the process and C the transfer function of the controller, the system can be represented by the loop transfer function $L = PC$ and the transfer functions

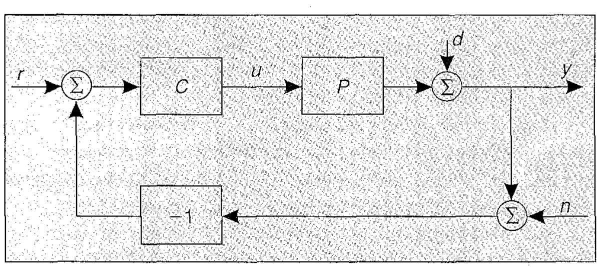


Fig. 5. Block diagram of closed loop system.

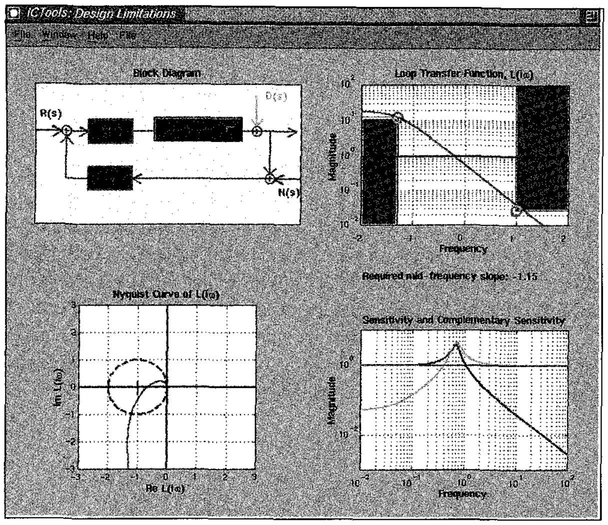


Fig. 6. Design limitations imposed by a non-minimum phase zero.

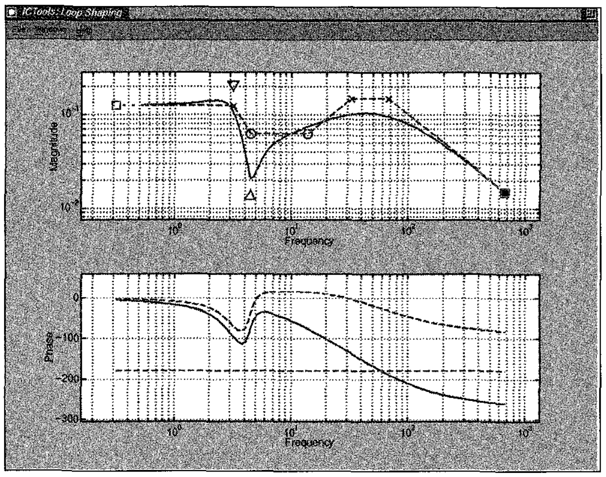


Fig. 7. A Bode diagram tool for manipulation of the break points with display of the phase curve and the minimum phase curve associated with the amplitude curve.

$$
\begin{aligned}
G_{er} &= -G_{en} = G_{yn} = S \\
G_{yr} &= -G_{ul} = T \\
G_{ur} &= -G_{un} = \frac{C}{1+L} \\
G_{yl} &= -G_{el} = \frac{P}{1+L}
\end{aligned}
\tag{6}
$$

where $S = 1/(1+L)$ is the sensitivity function and $T = 1 - S$ is the complementary sensitivity function.

Figure 6 shows a tool that illustrates fundamental design limitations in linear control systems. The question we address with this tool is: What should a good loop transfer function look like? The design of controllers that achieve a desired loop transfer function is addressed in a special loop shaping module.

The system is represented by four objects: a block diagram, Bode and Nyquist curves of the loop transfer function, and a Bode diagram of the sensitivity function and its complement. A schematic diagram of a linear system under unity feedback is shown in the upper left object. The loop transfer function is generated automatically from specifications of the low and high frequency gains. The specifications are altered by direct graphical manipulation in the upper right object. Adjustments of the loop transfer function are instantaneously reflected in the other objects on the screen. The generated loop transfer function has the form

$$
L(s) = M\frac{1}{(sT+1)^n}, \tag{7}
$$

where $1/T$ is the low frequency break point, M is the low frequency gain and n is a real number that ensures appropriate roll-off to meet the high frequency specifications. The performance of the closed-loop system can be assessed by inspection of the sensitivity and complementary sensitivity functions, as displayed in the lower right object. Using this tool, students quickly find the important characteristics of a good loop transfer function. A large value of the loop transfer function $L(s)$ gives T close to 1 and S small. It follows from Equation (6) that this ensures good command signal following, good rejection of load disturbances and small sensitivity to process uncertainty at low frequencies. To make sure that high frequency measurement noise does not propagate into the system, it must be required that the loop gain is small for high frequencies. (See Eq. 6.) Stability of the closed-loop system can be determined from the Nyquist plot of the open-loop transfer function, displayed in the lower left object. In this way, it is simple to investigate how narrow the mid-frequency band can be before the closed-loop stability is violated.

Using a pop-up menu, the loop transfer function can be augmented by a time delay, a non-minimum phase zero or an unstable pole. The limitations imposed by non-minimum phase elements can be clearly demonstrated in this way.

This tool demonstrates another strength of the interactive learning tools: Students with a good mathematical background can be motivated to further explore the consequences of Bode's sensitivity integral and Bode's relations, while students lacking this background can still develop an intuition even if they do not fully master the mathematical machinery.

Loop Shaping

Students can develop a good background for control systems design using the tools for process modeling, linearization and system analysis. Issues such as response times, steady-state errors, and robustness (in terms of different sensitivity measures) can be illustrated. A basic understanding of limitations in control systems design can also be obtained.

A wealth of control design methods has been developed over the years. Many ideas are based on loop shaping. The aim is to craft a compensator that modifies the loop gain of the system, so that the desired loop goals are attained. A good way to obtain a balanced view of trade-offs between different loop goals is to perform repeated controller designs. Normally, however, students only perform and evaluate a very limited number of control designs, mainly due to lack of time. By using an interactive tool, several iterations can be made with a minimal effort, and students can gain a considerable insight and experience. The interactive Bode diagram editor is a key component of such a tool.

The Bode diagram is convenient for quickly obtaining an overall view of the behavior of a linear system over a wide frequency range. Furthermore, a properly designed Bode diagram gives a clear indication of the non-minimum phase parts of a system. This is very important in order to make an assessment of design limitations. A Bode diagram consists of two objects, the amplitude curve and the phase curve.

In the amplitude curve, it is natural to display the asymptotes as well as the actual amplitude curve. The reason for this is that the asymptotes are directly related to poles and zeros of the system. The asymptotes are also easy to generate.

A given amplitude curve is uniquely related to a phase curve given by Bode's relation. For minimum phase systems, this curve is the actual phase curve of the system. If the system is not minimum phase, it may have larger phase lags, which cause severe limitations on achievable performance. This can be captured nicely by showing both the minimum phase and the actual phase of the system in the phase curve.

Let $G(s)$ be a transfer function and introduce

$$\begin{aligned} \log G(i\omega) &= A(\omega) + iB(\omega) \\ a(u) &= A(\omega_0 e^u) \\ b(u) &= B(\omega_0 e^u) \\ u &= \log\frac{\omega}{\omega_0} \\ \omega &= \omega_0 e^u \end{aligned}$$

where $A(\omega) = \log|G(i\omega)|$ and $B(\omega) = argG(i\omega)$. Assume that $\log G(s)/s$ goes to zero as s goes to infinity. The minimum phase curve B is then related to the amplitude curve through Bode's relations

$$\begin{aligned} B(\omega_0) &= \frac{2\omega_0}{\pi}\int_0^\infty \frac{A(v)-A(\omega_0)}{v^2-\omega_0^2}dv \\ &= \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{da(u)}{du}\log\coth\left|\frac{u}{2}\right|du, \end{aligned}$$

where $u = log\ v/\omega$.

An example of a Bode diagram tool is shown in Figure 7. It is natural to manipulate the Bode diagram through its breakpoints. To obtain agreement with the singularity diagram, breakpoints corresponding to poles are represented by 'x' while zeros are represented by 'o.' For complex poles and zeros it is necessary to specify the relative damping. For this purpose there is a special handle (marked by '∇') associated to each complex pole pair. To speed up the computation, only the asymptotes are changed during dragging. For rational transfer functions, the slope has integer values, so it is a simple matter to generate the asymptotes. The actual amplitude curve is updated when the mouse button is released. By double clicking on a breakpoint, the corresponding singularity is mirrored in the imaginary axis. Color is used to specify whether the poles and zeros are in the left or right half plane. In the phase diagram, we show the minimum phase curve and the actual phase curve. We can choose whether the gain is represented by the right or left end of the curve. The gain is changed by dragging either of the gain handles (marked by '□').

We have constructed a versatile loop shaping tool around the interactive Bode diagram editor. The tool allows interactive design of compensators. Frequency responses and time responses for a number of transfer functions, including those given in (6), can be displayed selectively. Apart from training basic loop shaping skills, this tool can also be used to explore many trade-offs that were disregarded in the design limitation tool. The information hiding feature is very useful in this context because it is possible to consider more details gradually as the students become more experienced.

Simple Controllers: Lead-Lag Design

Lead-lag compensation is a design method that is strongly graphic in nature. It is thus well suited for direct graphical manipulation. In principle, we could use the Bode diagram tool described above to make a lead-lag design. However, it is often advantageous to introduce special tools that are tailored to a specific purpose.

Proper parameterization is a key issue in the design of suitable man machine interfaces. One way to describe a lead-lag compensator is to use the following parameterization:

$$C(s) = kk_h\frac{(s+a_lk_l)(s+a_h/k_h)}{(s+a_l)(s+a_h)}.$$

The compensator is characterized by five parameters: the low frequency gain k_l, the mid frequency gain k the high frequency gain k_h, and the low and high frequency breakpoints a_l and a_h, respectively. Parameter k changes the overall gain of the compensator, and k_l and k_h represent the relative gains at low and high frequencies, respectively. This is illustrated in Figure 8.

It is natural to use the lag term to obtain good rejection of disturbances at low frequency properties. Since this is essentially a gain compensation, it is natural to use the gain curve primarily

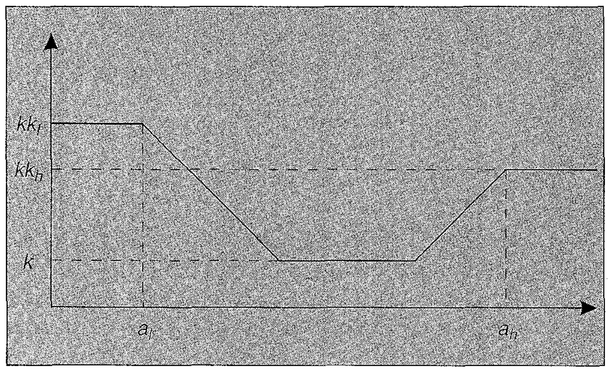


Fig. 8. Parameterization of the lead-lag compensator.

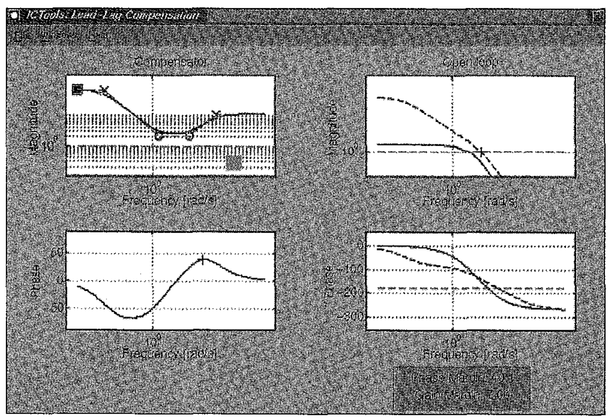


Fig. 9. Graphical user interface for the lead-lag design tool.

for lag compensation. Lead compensation is primarily done to obtain a good phase margin. It is therefore natural to achieve this by adjusting the phase curve. For this reason, we have provided a way to change the phase by direct manipulation of the phase curve. This is done by dragging the point at the frequency $\omega = a_h / \sqrt{k_h}$. This is approximately the point where the phase lead is maximal.

Figure 9 shows a graphical user interface for lead-lag design. The tool contains three objects. One shows the Bode diagram of the lead-lag compensator as described above. Another object shows the loop transfer functions of the compensated and uncompensated systems. The third object is an alphanumeric object that displays the amplitude and phase margins of the system. An evaluation of the design by simulations of the compensated and uncompensated system, for instance, is shown in a separate figure.

There are many issues about lead-lag design that can be learned from the tool. A common mistake, for example, is that the low frequency breakpoint is chosen too low. Since the low frequency gain is a good measure of low frequency disturbance rejection, it is easy to see the consequences of changing the break points. The fact that it may be quite critical to choose the lead compensator properly is also easy to demonstrate.

Student Responses

There is a long tradition of interactive computer tools for control education at Lund Institute of Technology. The particular tools discussed in this paper have been used since 1994. A computer based exercise based on the systems representation tool has been offered in the Basic Course. The exercise, which is voluntary, has been very popular among the students. About 60% of the 350 students who normally take the course usually sign up and do the exercise. Interactive computer tools are also used in courses on Computer Controlled Systems and Nonlinear Systems. A new approach in using interactive computer tools in the Basic Course was taken in the fall of 1997. New computer based self study exercises were introduced in the course plan, as a complement to the lectures, exercise sessions and laboratories. Four exercises were offered to the students: *Basic PID Control*, *System Representations*, *Pole Placement with PID Controllers* and *Compensation in the Frequency Domain*. Each exercise was composed of an interactive computer tool and a short written manual. The manual contains instructions for the tool, a minimum of theory, and suggested problems with solutions. The students were encouraged to "play" with the tools, and/or solve the suggested problems. All tools were designed to be self instructive, so that they could be useful also without the manuals. The exercises were offered for downloading from the Internet, but they were also installed on the workstations at the university. A sample of the students were asked to answer a questionnaire about the self study exercises at the end of the course. A total of 78 answers were received, and we believe that they are representative for the opinions of the 357 students who followed the course. It turned out that 88% of the students responded that the computer based self study exercises are good complements to the conventional teaching. Only 1% said the opposite (Some students did not answer this specific question, therefore answers do not total 100%.) As many as 85% used at least one of the tools. Among those students 24% responded that the tools substantially facilitate understanding of the theory, 73% said that they facilitate understanding to some extent, and only 3% found that the tools did not enhance the understanding. Moreover, 70% used the manuals, while 26% did not use the manuals. As many as 79% found the tools useful to "play around" with, without the manuals. None of the students found written manuals unnecessary. It was seen that 17% have worked with the exercises mainly at home, rather than at the school. The students were also asked to provide general comments. Some representative quotes are: *"Ok, now I understand more. It was fun." "... You made a great program!" "The exercises were in general made available too late." "Fun, more of this."* The student feedback will help us to improve the tools for future use.

To conclude, interactive computer tools have indeed been successful as self study exercises in the basic course. Even though the teaching format was new to both teachers and students, they have adopted it with enthusiasm. The student feedback clearly shows that the tools facilitate the understanding. Some students found the tools so helpful that they suggested making them mandatory. They also felt that the intuition provided by the tools made them perform better on the final exam.

A Renaissance of Classical Methods

Many of the classical tools for analysis and design of control systems are purely graphical. An advantage is that they give a very good insight into the properties of a system. Their main drawback is that they are cumbersome to work with and that it is difficult to obtain sufficiently accurate results. Recent developments in control system design has therefore been entirely focused on analytical techniques. However, it should be noted that manual tuning of weights are required even for analytical methods such as LQG-LTR and H_∞ loop shaping. This paper indicates that it may be possible to develop a new generation of design tools that combine intuitive graphical methods with analytical techniques. Loop shaping is a typical example where we can effectively iterate between manual interaction and optimization.

Conclusions

Experiments have shown that the time is now ripe for a new generation of interactive learning tools for control. The tools are based on objects which admit direct graphical manipulation. During manipulations, objects are updated instantaneously, so that relations between objects are maintained all the time. The tools are natural complements to traditional education, and allow students to quickly gain insight and motivation. A high degree of interactivity has been found to be a key issue in the design. Together with a high bandwidth in the man-machine interaction, this enhances learning significantly. Another nice feature is the possibility to hide minor issues and focus on the essentials. It is not easy to describe the power of these tools adequately in text. The best way to appreciate them is simply to use them.

We believe that there is a strong pedagogical potential for the type of tools that we have described. We are also of the opinion that we are only at the very beginning in the development of learning tools of this type. The addition of sound and animation are interesting avenues that should be pursued.

References

[1] K. J. Åström, "Computer aided modeling, analysis and design of control systems—A survey," *IEEE Control Systems Magazine*, vol. 3, no. 2, pp. 4–16, 1983.

[2] K. J. Åström, *Computer Aided Control Engineering, pp. 447–480. IEEE, Piscataway, NJ, 1995.*

[3] K. J. Åström and B. Wittenmark, *Computer Controlled Systems—Theory and Design,* 3rd ed., Prentice-Hall, Upper Saddle River, New Jersey, 1997.

[4] A. Blomdell, "Spread-sheet for dynamic systems—A graphic teaching tool for automatic control," *Wheels for the Mind Europe,* vol. 2, pp. 46—47, 1989.

[5] Boyd, C. Barrat, and H. Aling. *Interactive Control Design Module,* Integrated Systems, 1993.

[6] G. Crutchejeld and W. J. Rugh, "Interactive exercises and demonstrations on the web for basic signals, systems, and control," *Proceedings of the 36th IEEE CDC,* San Diego, CA, December 1997.

[7] M. Gäfvert and M. Johansson, "Ictools — interactive learning tools for control," *http://www.control.lth.se/~ictools*, Department of Automatic Control, Lund Institute of Technology, Lund, 1997.

[8] E. Granbom and T. Olsson, "VISIDYN — Ett program f6r interaktiv analys av reglersystem," (VISIDYN—An interactive program for design of linear dynamic systems), Master Thesis TFRT-5375, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, November 1987.

[9] H. Haglund, "Dynamic pictures in sampled data systems." Master thesis ISRN LUTFD2/TFRT- -5542- -SE, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, November 1995.

[10] R. 0. Harger, "Introducing DSP with an electronic book in a computer classroom," *IEEE Transactions on Education,* vol. 39, no. 2, pp. 173–179, May 1996.

[11] M. Jamshjdi and C. J. Herget, *Computer-Aided Control Systems Engineering,* North-Holland, Amsterdam, 1985.

[12] L. Jia and W. Schaufelberger, *Software for Control Engineering Education,* Verlag der Fachvereine an der EHT-Zurich, Zurich, 1995.

[13] M. Johansson and K. J. Åström, "Generalized spread-sheets for CACSD," *Proc. of The IEEE Int. Symp. on CACSD*, Dearborn, Michigan, September 1996.

[14] M. Johansson and K. J. Åström, "Virtual interactive systems for control education," *Proc. 35th IEEE Conf. on Decision and Control,* pp. 3888–3889, Kobe, Japan, December 1996.

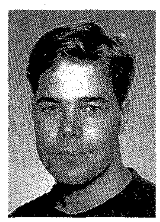
[15] D. A. Linkens, *CAD for Control Systems,* Marcel Dekker, New York, 1993.

[16] J. Pagel, Y. Sun, D. Tilbury, L. Oms, M. Suri, and W. Messner, "Control tutorials for Matlab on the world wide web," in *Proc. of the American Control Conf.,* Albuquerque, NM, June 1997.

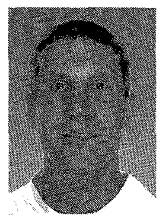
[17] O. Ravn, "On user-friendly interface construction for CACSD packages," *Proc. of the 1989 Control Systems Society Workshop on ComputerAided Control System Design,* pp. 35–40, Tampa, FL, December 1989.

[18] C. A. M. Serrano, "Dynamic picture as a learning tool in control," Master Thesis ISRN LUTFD2/TFRT- -5566- -SE, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, September 1996.

[19] B. Wittenmark, H. Haglund, and M. Johansson,"Dynamic pictures and interactive learning," *Control Systems Magazine, vol. 18, no. 3, 1998.*



Mikael Johansson received his M.Sc. in Electrical Engineering from Lund Institute of Technology in 1993. Since 1994, he has been affiliated with the Department of Automatic Control at Lund Institute of Technology, where he is currently working on his Ph.D. degree. His research interests are in analysis and control of nonlinear and hybrid systems.



Magnus Gäfvert received the M.Sc. degree in Engineering Physics from Lund Institute of Technology in 1996, where he is currently pursuing a Ph.D. degree in Automatic Control. His research interests include issues on distributed control systems, and nonlinear control with focus on friction modeling and compensation.

Karl Johan Åström has been Professor of Automatic Control at Lund Institute of Technology since 1965. He has a broad interest in automatic control including, stochastic control, system identification, adaptive control, computer control and computer-aided control engineering. Astrom has supervised 42 Ph.D. students, he has coauthored five books and many papers. He has served IFAC in many capacities. Astrom is a Fellow of the IEEE, a member of the Royal Swedish Academy of Sciences (KVA), the Royal Swedish Academy of Engineering Sciences (IVA), and the US National Academy of Engineering. He has received several awards among them the Giorgio Quazza Medal from IFAC in 1987 and the IEEE Medal of Honor in 1993.