Guidelines for Designing Better Motion Control Systems

Lee Stephens, Kollmorgen

Introduction

Motion control system engineers work chiefly in two well-defined areas: 1) new designs and 2) redesigns—or retrofits. Of course, the new design task starts with a clean sheet of paper. Since the system did not exist before, all the specifications must be derived from calculations, simulations or actual measurements on existing machinery. They can measure feeds, speeds, loads and torques on similar equipment that operates without servo controls, or they can simulate them with a variety of software packages.

Design or Redesign? What's Best?

Some might expect that designing a new system would be more difficult than replacing an old one, but this is often not the case. Although the loads, speeds and torques needed might be known, a newer digital controller that replaces an old analog system using brushed DC motors behaves differently. New control laws often enter the equation, and when the designer is not aware or does not anticipate these differences, the first system off the drawing board might not live up to expectations.

One major factor to consider in the new system is calculation time. A digital system works in three serial, quantifiable steps—i.e., measure, calculate and output. The controller requires specific time slices to run through these three functions. The calculation period might be so long as to let the system wander out of control. An analog system does not have this particular drawback to the same degree. By comparison, the analog system calculates, measures and outputs almost simultaneously and continuously. Typically, the lag time is not as severe as the calculation delays of a sampled system.

Both new and replacement systems follow the same basic laws of physics—but different control laws—so the design approach and hardware shopping list could be very different for each system. For example, a new system design can be defined in two ways.

The first is straightforward, where the controls engineer designs a system totally on his own from the ground up. He completely defines the system and orders the components needed to do the job. He alone is responsible for the outcome.

Or, a new system might involve a client that has a resident engineer who helps define the system parameters and selects the components. The consulting motion controls expert may help design the client's new system after its resident engineer had already selected a few key components. The resident engineer may have determined loads, speeds and torques from actual measurements, calculations or simulations based on a few assumptions. He also may have purchased some major components—motors and transmissions, for example, based on these determinations—before hiring the consultant. The consultant's initial posture is to assume that the components that the resident engineer selected are perfectly suitable. Unfortunately, this sometimes is not the case. Assumptions may have been made under static conditions, when they should have been dynamic—particularly regarding the load. The result is that the consultant now has no choice but to revise the model to include the proper parameters.

Strategy

Calculation delay is especially troublesome when a control system contains multiple axes. In a three-axis, pick-and-place system, for example, the *x*, *y* and *z* axes all must converge simultaneously on a particular point. If a lag appears in any one or more axes, the component part could be set in the wrong place or delayed long enough to affect throughput of the machine. To avoid this, first determine the bandwidth of the system. Measure the load inertia, determine how fast it must move, and, more importantly, how fast it needs to settle. Settling time really dictates the bandwidth. Bandwidth does not determine the speed; it determines how quickly and precisely the load stops or follows a contour.

The bandwidth is usually defined in terms of its –3 dB point and the 45° phase shift. Don't prefer one parameter over the other; exceeding the – 3 dB point and the 45° phase shift indicates that the system is out of control. For example, if the closed-loop system is at –3 dB but it has a 60° phase shift, it was significantly out of control long before it hit the –3 dB point.

In digital systems, other functions affect the phase angle, which sometimes surprises the customer. It concerns current-loop bandwidth; it is calculated digitally and the calculation delays become apparent. Although the delay is worse than a phase shift, it is essentially the same thing; i.e., phase is time in the frequency domain. The digital system may not have optimized control algorithms to calculate current, velocity or position. It is then that the digital system may have more of a calculation delay than the system can tolerate; it did not achieve the intended bandwidth.

High-performance drives offer sample rates for current and velocity that are generated in the field-programmable gate array (FPGA—*an integrated circuit that is configured by a user or designer post-manufacture; i.e., field-programmable*) that minimize the effects of sample and hold errors and the calculation delays since the FPGA is operating much faster than would a typical processor. The combination of digital signal processing—DSP and the FPGA technology are a step ahead of standard processing —(*DSPs take real-world signals like voice, audio, video, temperature, pressure, or position that have been digitized and then mathematically manipulates them. A DSP is designed for performing mathematical functions like "add," "subtract," "multiply" and "divide" very quickly.* Source: Analog Devices, Inc.). Performance now approaching the analog systems has been achieved, making replacement and retrofit easier.

Modeling

Some of the design work is carried out by modeling with a variety of software packages. Systems may be modeled in a digital format to determine not only the bandwidth but also the position accuracy. The model can become as detailed as necessary. Widely used software packages include *VisSim*, *LabView*, *Mathcad*, *Matlab* and *Motioneering*, the latter available from Kollmorgen.

VisSim is a widely used simulation software package; a trial version may be downloaded free of charge from their Web site. It can run and modify the model but it cannot be saved in the limited version. *LabView,* from National Instruments, can be used to model as well as control an EtherCat-based drive; this becomes valuable for breadboard applications. *Motioneering* is free and determines how much current and power the motor needs to function properly. *Matlab* has some options available like *Simulink* to help modeling control systems. This is also the basis of the *Mechaware* models within the *SynqNet* control systems from Kollmorgen. *Mathcad* is another mathematic-based modeler that many designers use.

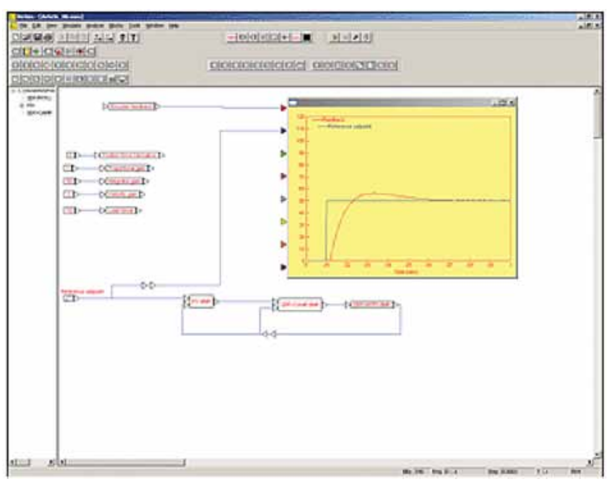

Figure 1—Screen simulation of a PIV control system response using *VisSim* modeling software. The ideal step-response curve or set point shown in black is compared to the feedback response signal (in red) when the system is programmed with constants for proportional gain of 1.0, integrator gain of 50, velocity gain of 0.1 and a load force of 10.

When modeling, it can be difficult to decide when to stop. As designers gain sufficient experience, however, they can recognize when some parameters are not relevant enough to consider. If you start with a basic motor/load-with damping model, you then add the parameters that are generally relevant to the control system. In the beginning, try everything on the list, then narrow it down to the few parameters needed to adequately and sufficiently model the system. Any adjustments after that are usually minor. Stop when further detailing does not make the model any better or does not gain any more advantage in the design. Often, designers return to the model after

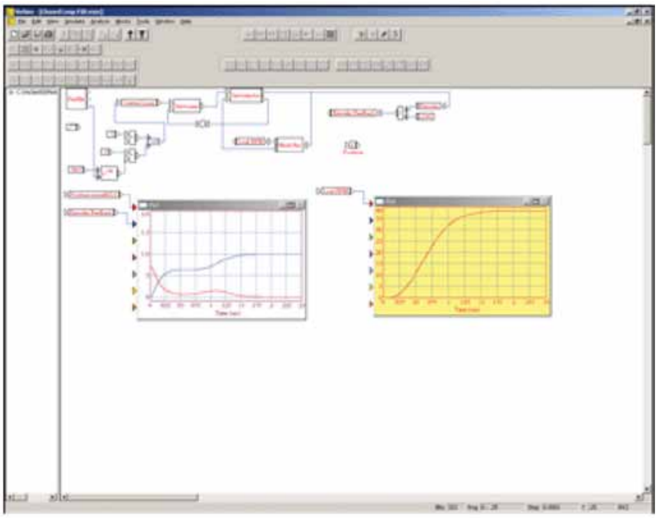

Figure 2—Simulation illustrating a screen shot of a block functional diagram of a closed-loop system. The left-hand graph depicts the position error in red and the encoder feedback in black. The load speed (rpm) vs. time is shown in the right-hand graph. The result of different gain, acceleration and velocity constants can be seen as they are changed.

the system did not meet specifications, and they find errors; they go through the system and might find that they should have considered something else that was more important. Always try to validate the model, as it is only as good as the information that goes into it. Some information is difficult to determine mathematically and has to be done empirically. However, the empirical information determined for the model may not be sufficient, so use multiple formats including the frequency domain and time domain. *VisSim* and *Mathcad*, for example, work well together for this, so take advantage of them.

Validation

After the model is completed, the hardware may not yet be available. The customer may have the breadboard and prove that the supplier's products will work in his system, but the consultant may not be in control of how close the breadboard represents the real system. He may use the model as supplied, or he may revise it and find changes in pulley ratios, motors, inertias or miscalculated inertias. However, assume the physical breadboard is sufficient to model and validate the system. In the next step, the customer now implements the alpha or beta stage of the project and buys the needed components.

On occasions when the system does not meet expectations, sometimes the reason is a miscalculation. The system was expected to operate under certain circumstances, but perhaps could not. The model should have shown how much headroom there was, and how close it was. If not, then it may be necessary to construct a Bode plot or run a Fast Fourier Transform (FFT) algorithm to find out why the system is vibrating, resonating or not rejecting the disturbance.

The ability of a control system to reject disturbances is a figure of merit that goes beyond a number; it is intuitive. It is through feedback that the disturbances will be rejected. The proper feedback loop will let the load reach the intended position. This might involve velocity, position or torque feedback. The nature of the feedback loop depends on the function that is specifically needed, such as torque or current for machining operations.

During the build stages (breadboard, prototype, alpha, beta and first piece) the model should be visited to verify and modify accordingly to the empirical data or corrections to the physical system for validation. The largest pitfall can be allowing the model to become obsolete with empirical tests; in later stages of development, problems could arise where the model can supply useful data. This is the tool for validating each stage of the project with the original specification and intention.

Feedback

Feedback system complexity varies with the application; it can be as simple as an incremental encoder or a resolver, and there are different reasons for using one or the other. A resolver is extremely robust and can tolerate harsh environments well. It can be a sine/cosine encoder, which can handle very high bandwidth, but typically, does not do well in a high-vibration, high-temperature environment.

Some systems become more complex when a secondary feedback device is needed. For instance, a system may require a feedback device mounted directly on the motor, and another closer to the load. The feedback device on the motor could be used for velocity control, but the feedback device on the load would be used for final position. It doesn't sound too difficult, but as the feedback device moves farther from the transmission device—the motor, in this case—the more items will be enclosed in the loop. Another example: when a resonance appears at a certain

Figure 3—A *Motioneering* model for a typical PID controller lets designers examine the system behavior with numerous gains and other parameters. It can show a plot of this behavior with certain perturbations.

frequency that is outside the intended control bandwidth, having the feedback device outside of the load is certainly going to help move the load to its final position. But tuning this system—i.e., getting the position system up to the desired performance level—is extremely difficult. Therefore, when a very high bandwidth position loop is needed, a secondary encoder is typically placed at the end of the position loop. Elimination of the mechanical transmission (belts, gearboxes) through the use of direct drive technology is also a potential solution, enabling easier tuning as well.

Redesign

Occasionally, during tuning, the controls engineer will find that rather large modifications must be made to the system to gain the intended control. He may even have to change his entire theory regarding the method needed to tune the system.

The simplest control method—either PID (proportional, integral, derivative) or PIV (proportional, integral, velocity)—is typical in a great number of systems. A standard PID control system is relatively easy to tune; it can be used for either a new system or an old one. Here, the damping is a derivative of position error. The proportional term is a gain factor. In PIV, the velocity is calculated and used for the damping of the system.

Other control laws exist and may be a good selection. Kollmorgen's *MechaWare* allows custom algorithms and filters that can accommodate the most stringent of applications and needs. Custom filters generated from the four individual bi-quad filters in the aforementioned AKD servo drive should minimize redesign challenges.

Stiffness

System stiffness—or a lack of it—continues to be a major, chronic problem. Say, for example, resonance problems indicate that a system is not stiff enough. And backlash in a system is another serious problem. Here, the customer may have a linear motion control system that specifies a rack-and-pinion transmission on a precision axis. However, such a single-format gearing system produces troublesome backlash; not even anti-backlash gears can solve the problem. They often still have enough backlash to create instability and typically contain two gears in an interference fit; some friction losses are inevitable. That arrangement does not guarantee zero-backlash—only that the backlash is taken up by another mechanism. They have a spring rate to contend with, which is a dilemma when trying to control a frequency in the domain of the spring rate. It is a common problem, usually found in a system designed by someone lacking controls experience.

To overcome these problems, conduct Bode plots and La Place transforms in the frequency domain. Compare the Bode plot performance with the La Place transforms and tune the system based on that information. Observe the frequencies, disturbances and amplitudes, and then determine the best method of attack to eliminate the disturbances or insert compensation to reject them. In addition, stiffen the system to eliminate resonances and raise the frequencies above the frequency of disturbance. Also, at times the system may be damped, but this could also affect the compliance. High-frequency damping usually does not add compliance; but at low frequencies, damping certainly cannot be used because it adds compliance that exacerbates the disturbance itself. Try using acceleration feedback, a Lowenburger observer (a relatively complex algorithm) or select a suitable filter.

Conclusion

With a "living" model documented, updated and validated, a nearly seamless path from concept to final product can be achieved. This process can only be guaranteed with periodic updates, validation to empirical data and proper use of tools for modeling. Good vendor data and upfront modeling minimize surprises and unexpected—or unwanted—results. The motion control vendor should have experience in providing accurate data on the products and rudimentary product information should it be required. A good quality vendor will alleviate many of the pitfalls.

Lee Stephens *possesses an engineering science degree from Broome Community College in New York. Stephens has 25+ years experience in the semiconductor industry—primarily in motion control—and has spent the past nine years as senior motion control engineer/systems support, for Kollmorgen. He has written and published numerous technical articles on various motion control topics.*