

Comparison of Motion Control Loops for Industrial Applications

George Ellis
Kollmorgen Corporation
201 Rock Road
Radford, VA 24060
email: GEllis@Kollmorgen.com

Robert D. Lorenz
University of Wisconsin-Madison
Depts. of Mech. Engr. and Elect. & Comp. Engr.
1513 University Avenue
Madison, WI 53706
email: Lorenz@eceserv0.ece.wisc.edu

ABSTRACT - High-performance AC and DC industrial servodrives use standardized motion control algorithms. The algorithms are based on common feedback sensing methods (digital position measurement via encoders or resolvers) and a common assumption that the electromagnetic torque dynamics are substantially faster than the motion control dynamics desired. The vast majority of these motion control algorithms close the motion control loops in one of two ways: 1) an average velocity loop is cascaded with a position loop, or 2) multiple state variable loops are closed in parallel. If the state variable form is properly configured, the command tracking properties are virtually independent of the disturbance rejection properties. However, this controller would require a command for acceleration which is frequently not available. In this case some modified form of cascaded loop controller topology is often used. In this paper the alternative methods for this case are explored and tuning guidelines developed based on both simulation and laboratory results.

I. BACKGROUND

High-performance AC and DC servodrives are widely used in motion control applications such as machine tools, packaging, printing, web handling, robots, textiles, and food processing. The motion control algorithms are based on the mechatronics assumption of nearly ideal electromagnetic torque control. This assumes ideal field orientation and current regulators of bandwidth considerably beyond the motion control bandwidths desired. Feedback devices, chiefly encoders and resolvers, are employed in these systems to sense motor position and to calculate the sample average motor velocity, albeit with significant average velocity resolution limitations. The vast majority of motion control algorithms employed in industrial applications are of two forms: 1) an average velocity loop is cascaded with a position loop, or 2) state variable loops (Proportional-Integral-Differential, PID position loops) are closed in parallel.

The state variable loops in parallel (PID position) configuration is known to completely separate command tracking tuning from disturbance rejection tuning [6,7,8,11,12]. However, such motion controllers require an acceleration command which is often not provided in industrial servodrive controllers. This causes the tuning to again become cross-coupled and interdependent.

The cascaded loop topology is very commonly found in industrial servodrives and has a variety of adjustments to handle this (unwanted) cross-coupling of the tuning process.

The cascaded average velocity loop is usually a Type I

(integrating) loop which is cascaded with a proportional position loop. In that case, there are two types of average velocity loops that are commonly employed: Proportional-Integral (PI) and Pseudo-Derivative Feedback (PDF) [1,3,4,5]. Both of these controllers offer different possibilities for handling the tuning cross-coupling.

This paper focuses primarily on an alternative to PI and PDF which will be called here $PI+$ but is also sometimes called PDFF. $PI+$ will be shown to be a general controller within which PI and PDF are special cases. Further, $PI+$ will itself be shown to be a special case of the PID position loop. This paper will present simulation and laboratory experiments for the $PI+$ controller. The goals of this paper are to provide a quantitative analysis of PI, PDF, and $PI+$, and to compare the methods to the PID position controller for the case where no acceleration command is available.

II. VELOCITY LOOP ALTERNATIVES

In cascaded motion control systems position profile generators provide a position command and also an average velocity command to support so-called "velocity feed-forward." The average velocity loop output feeds a cascaded current regulated field oriented drive which creates electromagnetic torque in the motor. Position is fed back from a position sensor. Velocity is not measured. Currently most systems calculate sample average velocity from the difference of the two most recent positions divided by sample time, T , as shown in Fig. 1.

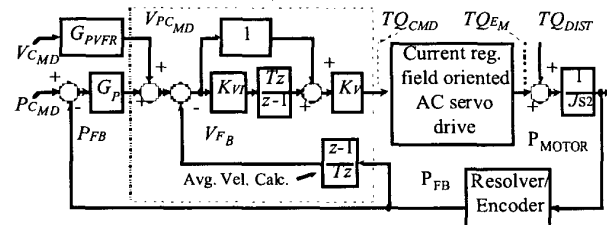


Fig. 1. PI-velocity loop cascaded with position loop.

A. PI Velocity Loop

PI loops are the most common velocity loops today. The velocity error integral is scaled by K_{VI} and then added back to the error. The sum is scaled by K_V . In some cases, the integral and proportional terms are scaled independently and then added. These two methods are equivalent although the units for K_{VI} change. The output equation for PI is:

$$TQ_{CMD} = (K_{VI} \int (V_{PCMD} - V_{FB}) dt + V_{PCMD} - V_{FB}) K_V. \quad (1)$$

Note that for PID velocity loops, a term would be added that scales the "derivative" of the average velocity error. PID velocity loops are rarely used because calculating sample average acceleration from average velocity suffers from resolution limitations so severe as to make it impractical.

Note that PID-position control, which will be discussed below, is quite different from PID-velocity control. For example, the D term of a PID velocity controller is proportional to acceleration whereas the D term of a PID position controller is proportional to sample average velocity.

B. PDF Velocity Loop

A second velocity loop frequently used in cascaded-loop systems is PDF [1]. The velocity error integral is scaled by K_{VI} and then added to the negative of velocity feedback. The output is scaled by K_V . The output equation for PDF is:

$$TQ_{CMD} = (K_{VI} \int (V_{PCMD} - V_{FB}) dt - V_{FB}) K_V. \quad (2)$$

PI and PDF are similar; they share the integral (K_{VI}) and output (K_V) scaling terms. The difference is that the second V_{PCMD} term in PI (1) is eliminated from PDF(2). This makes PDF less responsive to the velocity command than PI. However, the change in structure allows PDF to have higher integral gains while avoiding overshoot to otherwise infeasible (step change) motion commands. This trade-off allows better response to low-frequency disturbances [2].

C. PI+ Velocity Loop

PDF can be augmented with a feed-forward term to create PI+. The feedforward term, scaled by K_{VFR} , injects the command ahead of the integral making the system more responsive to commands. The output equation for PI+ is

$$TQ_{CMD} = (K_{VI} \int (V_{PCMD} - V_{FB}) dt + K_{VFR} V_{PCMD} - V_{FB}) K_V. \quad (3)$$

The block diagram is shown in Fig. 2.

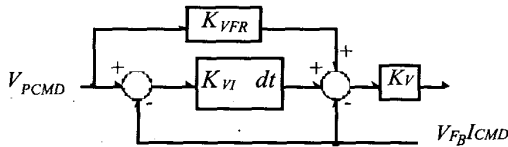


Fig. 2. PI+ velocity controller

PI+ is a general controller which includes PDF and PI control (when K_{VFR} is 0 and 1, respectively). While K_{VFR} can be set to these extremes, it can also be set anywhere in between. One area addressed in this paper is optimizing the value of K_{VFR} for an application.

A more detailed and complete motion control structure based on PI+ velocity control is shown in Fig. 3. This system will be used to generate data later in this paper; it includes the effects of sampling at 250μs, encoder resolution (1024 lines), and the z-domain representations of

"integration" and "average velocity calculation".

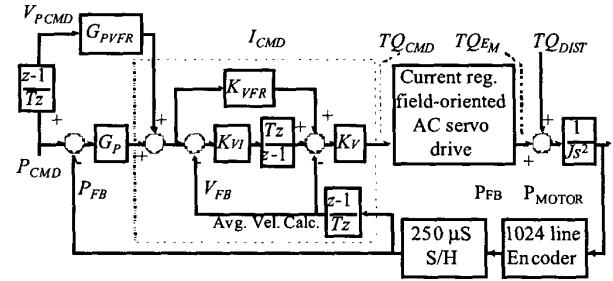


Fig. 3. Cascaded loop motion controller based on PI+.

III. STATE VARIABLE CONTROL DESIGN BASED ON PHYSICAL PROPERTIES

The controller of Fig. 3 can be redrawn in a state variable format as shown in Fig. 4.[3-7] This form is commonly referred to as "PID position control" in industry. The state format is equivalent to Fig. 3, where:

$$B_A (\text{Nm/rad/sec}) = K_V, \quad (4)$$

$$K_{SA} (\text{Nm/rad}) = K_V (G_P K_{VFR} + K_{VI}), \quad (5)$$

$$K_{IA} (\text{Nm/rad/sec}) = K_V G_P K_{VI}, \quad (6)$$

$$K_{VF} (\text{Nm/rad}) = -K_V (1 - G_{PVFR} K_{VFR}), \quad \text{and} \quad (7)$$

$$K_{PF} (\text{Nm/rad/sec}) = -K_V K_{VI} (1 - G_{PRFR}). \quad (8)$$

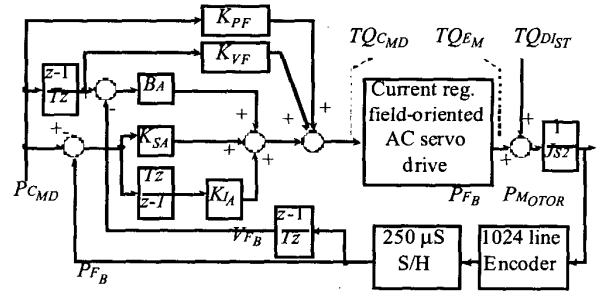


Fig. 4. Cascaded loop controller based on PI+ in state variable form.

One of the primary benefits to the state variable format is that the state feedback gains all have physical meaning and physical units. Thus K_{SA} has units of static stiffness (Nm/rad). This means that this controller gain can be checked in situ by purely mechanical means. In addition, the dynamic stiffness of the drive is now explicit. If the current loop, sampling effects, and resolution limitations are ignored, dynamic stiffness can be written by inspection:

$$\frac{TQ_{DIST}(s)}{P_{MOTOR}(s)} = \frac{1}{Js^2 + B_A s + K_{SA} + K_{IA}/s} \quad (9)$$

The frequency response plot for dynamic stiffness is shown in Fig. 5:

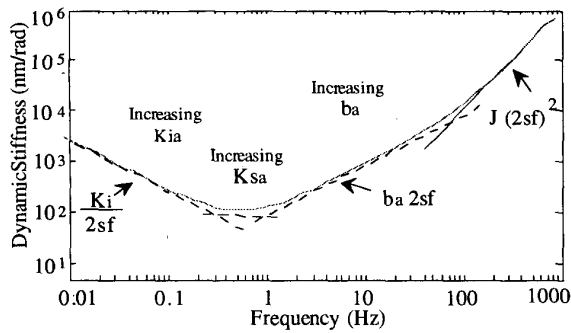


Fig. 5 Dynamic stiffness of the state variable motion controller

Command tracking for the state variable controller can be achieved independent of loop gains if both a full command vector (position, velocity, and acceleration) is available and the command trajectory is physically feasible [6,7,8,11,12]. If a reduced command vector is used, the command tracking will be limited to the loops for which state commands are used.

The industrial motion controller of Figs. 3 and 4 has only a position and velocity command. It lacks the acceleration command term. In many industrial applications acceleration is simply not provided. Additionally, in so-called *master-slave* applications, the average velocity command signal is derived from a position sensor so that the velocity command is too noisy to differentiate. While solutions to these problems have been developed [8,9,11,12,13] they are not commonly available in off-the-shelf industrial servodrive products.

The working engineer is then required to tune the control system without the full command vector (without acceleration command) for command trajectories that may or may not be physically feasible. If the application requires maximum dynamic stiffness, then the gains B_A , K_{SA} , and K_{IA} (or their equivalents) must be maximized. Unfortunately, without the acceleration command, undesirable overshoot can occur in response to even a common, trapezoidal velocity command which should be feasible.

Given this general lack of an acceleration command, the next section will address optimal selection of tuning gains of the PI^+ controller when the full command vector (acceleration command) is not available.

It should be noted, that selection of feedforward gains (K_{PF} and K_{VF}) in the PID position control to avoid overshoot is not intuitive when no acceleration command is present. Ideally, if the acceleration command were present, they would both be zero. However, when no acceleration command is provided, their optimum values for a given feasible command trajectory are functions of loop gains. Furthermore, they are negative for most applications in order to reduce the tendency to overshoot when the loop gains are high. If these feedforward gains are set to zero (as would be done if acceleration command were used), the feedback gains K_{SA} and K_{IA} must be limited to reduce the overshoot.

IV. EXPERIMENTAL RESULTS

In order to analyze the PI^+ controller, three items were required: 1) A control system appropriate for industrial motion control that implements the PI^+ algorithm, and that has on-board position loops and a profile generator; 2) Instrumentation that can measure time- and frequency-based response of the drive system; and 3) a verified model of that system.

The physical unit is required to validate results of the model. The on-board position loop and profile generator simplify experimentation by locating the entire control system in the drive. The model is used to evaluate dynamic stiffness, which can easily be measured on the drive, but only within the dynamic limits of the current loop. The model of course does not have this limit. The Kollmorgen SERVOSTAR® AC drive amplifier and Kollmorgen GOLDLINE® PM AC servomotor were selected as they met the requirements. The model of Fig. 3 was used with one change: scaling constants were added so that the terms in the model (K_{VFR} , etc.) matched the drive. Also, the PWM current controller model (2-pole low-pass filter with $\xi = 0.707$, $\omega_N = 800$ Hz) was experimentally verified.

The model was built in a time-based modeling environment called ModelQ. A stand-alone executable version of this model is available at no charge from the first author (e-mail gellis@kollmorgen.com.) The ModelQ environment provides time-based and frequency-based information to allow thorough evaluation of the simulated system.

The model of Fig. 3 was verified using a motor/drive system with measurements from an oscilloscope internal to the drive, and a SigLab Model 20-42 frequency analyzer by DSP Technologies. A photograph of the test system is shown in Fig. 6. The drive and motor set are on the left; the frequency analyzer is beneath a laptop computer.



Fig 6. Photograph of Test Setup

The tuning constants for a typical velocity loop were entered into the drive and into the model. The closed-loop command response was measured from the actual system and

from the model; then the two were compared. The gains of the two plots did not differ by more than a few tenths of a dB below 100 Hz; the phase did not differ more than two degrees over that same range. Between 100 and 200 Hz, gains were equal within about one dB and phase was accurate to within 350 μ sec. An adjustment of 8% was made in the motor torque constant (K_T) to achieve this correlation. The adjustment corrects for normal variations in motor K_T and in amplifier current scaling. Command tracking frequency response plots were run for several sets of tuning values with the same results.

The command tracking frequency response of the actual system is shown in Fig. 7; gain and phase are shown on the same graph with units listed to either side of the graph.

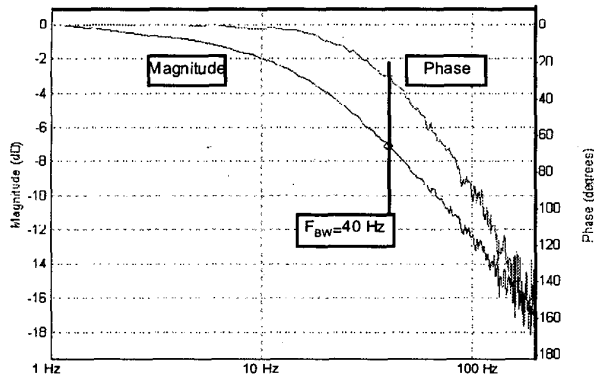


Fig 7. Command tracking frequency response of actual system

The command tracking frequency response for the model system is shown in Fig. 8 in two frames with the gain above and the phase below. At this point, the model was validated and was used for many of the system measurements.

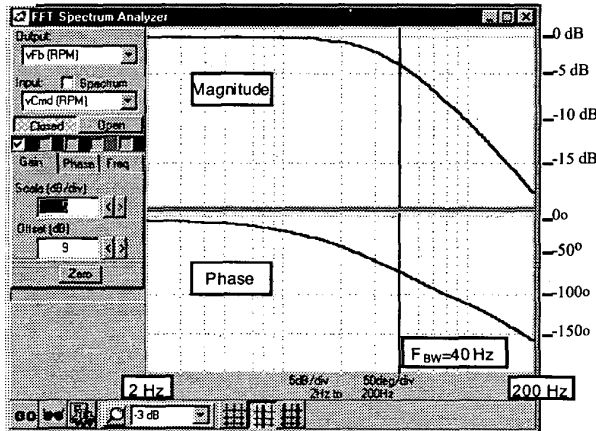


Fig 8. Command tracking frequency response of modeled system

A. Performance Criteria

The next step in the comparison was to establish the performance criteria by which the velocity loop would be measured. The criteria were selected based on experience with industrial applications:

1) Response to command

The time-based (step response) and frequency-based (bandwidth, Bode plot) of the system command response is measured. Higher responsiveness is frequently an important factor in motion systems.

2) Noise

System generation of acoustical noise and physical movement of the motor (when commanded to zero speed) were observed on the actual drive.

3) Dynamic stiffness

The ability for the system to create torque in response to a broad-band frequency disturbance on the motor shaft was evaluated from the system model.

4) DC following error

In most positioning systems there is a difference in the position command and feedback when the motor is rotating at constant speed. This is often called DC following error. For many applications, DC following error should be minimized.

5) DC Stiffness

The ability for the system to create torque in response to a very-low-frequency disturbance on the motor shaft was evaluated from the system model.

6) Resonance

Resonance is the tendency of the motor/drive to oscillate because of a compliant coupling between the motor inertia and the load inertia. This area of motion control is studied frequently because resonance is common in industrial applications.[7,13,14] Response to compliance was measured on the physical system.

7) Insensitivity to variation of inertia

In some applications, the inertia of the load may change as in the case of wind/unwind applications, or because the geometry of the machine may vary the inertia reflected to the motor as occurs in many types of robots. The less sensitive to inertia the method is, the better. Sensitivity to inertia was evaluated from the system model and then confirmed on the working system.

B. Method of tuning

All comparisons were based on the cascaded position controller of Fig. 3 with no acceleration command. The position control configuration has five tuning constants (G_P , G_{PVFR} , K_V , K_{VI} , and K_{VFR}) which provide too many degrees of freedom to allow independent variation of all constants. A procedure was required to generate sets of constants based on objective measures of performance. This procedure was based on laboratory tests and industrial experience. The constants were set as follows:

1) Selecting G_{PVFR}

G_{PVFR} is the scaling constant for average velocity feedforward (FF) signal generated from the position command

trajectory. This feedforward term is used to increase responsiveness and reduce following error. When G_{PVFR} is 100%, all DC following error is canceled. However, with no acceleration command, the system overshoots so much that this setting is considered impractical for most applications. With no acceleration command, a setting of 75% is considered high since it induces enough overshoot that the system gains must be reduced. Setting G_{PVFR} to 0% turns off velocity FF and allows the highest feedback gains (highest dynamic stiffness) without inducing overshoot. However, systems with low G_{PVFR} have more following error. For this paper, two sets of experiments were run: one with 75% G_{PVFR} and the other with 0%. G_{PVFR} does not affect the stability.

2) Selecting K_{VFR}

Numerous values of K_{VFR} were used including: 0%, 25%, 50%, 60%, 75%, and 100%.

3) Selecting K_V

Investigations in the laboratory and with the model indicated that two factors of performance criteria were impacted almost exclusively by K_V : noise generation from resolution [8,9] and susceptibility to mechanical resonance. Noise was evaluated on the physical unit. The acoustical noise and shaft rotation of the test system is most directly influenced by K_V .

Mechanical resonance is caused by a compliant coupling between motor and load. A model of a two-bodied mechanical system is shown in Fig. 9. The compliant model includes a spring constant between motor and load (K_S), a cross-damping term between motor and load (K_{CV}) for losses. The model does not include the effects of Coulomb friction. The compliant load is used in the system model.

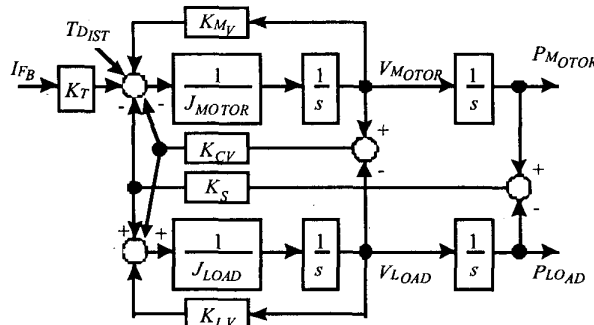


Fig. 9. Compliant load/motor couplings

Reducing susceptibility to resonance is a complex problem and one that is extensively studied [4,9]. Susceptibility was influenced almost wholly by K_V (larger K_V causing greater susceptibility). This was tested in the model and verified on the actual system using a variable compliance load as shown below in Fig. 10. This load consisted of a shaft and a movable inertia. The farther the inertia from the motor, the more compliant the coupling. The inertia was moved to various positions and then the system tendency to oscillate for various values of tuning constants was evaluated.

After some experimentation K_V was chosen as 3000. This kept two measures of performance, noise generation and susceptibility to resonance, equal between all constant sets.

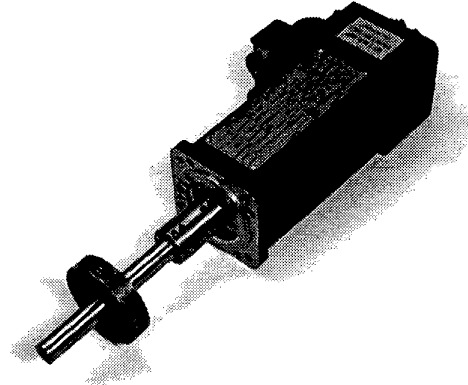


Fig. 10. Motor with variable compliance load

4) Selecting G_P and K_{VL}

G_P and K_{VL} were adjusted to maximize response within a given stability criterion. The criterion was based on a 50 mSec trapezoidal move shown in Fig. 11 with velocity command above and feedback below. This type of move is common in high-performance applications: it is short (50 mSec) and follows the traditional 1/3-1/3-1/3 profile where the acceleration, deceleration, and traverse time periods are equal. This profile is the best straight-line fit to parabolic profile which uses minimal energy to rotate an inertia in a fixed period of time.

The stability criteria allowed only 0.5% overshoot and ringing. For all data sets there was a unique set of G_P and K_{VL} which just met this criteria.

V. DATA COLLECTION

Data was collected in two categories: command response and dynamic stiffness. In both cases, time-based and frequency-based data was taken from the simulation. All data is listed in Tables 1 and 2 which are listed after the references in this paper.

A. Command response

Command response was measured according to bandwidth (-3dB frequency), phase lag, and response to the trapezoidal velocity profile shown in Fig. 11. Dramatic improvements in command response were realized with increasing K_{VFR} .

B. Bandwidth

The bandwidth is the frequency where, when the system is excited with a small-signal sinusoid, the response falls to 70% (-3dB) of the low-frequency response. Note that the bandwidths listed in Tables 1 and 2 are the position loop bandwidth. The results were that bandwidth improved with more K_{VFR} by 30% to 50%.

C. Delay at 20 Hz.

The time delay between the command and the feedback when the excitation is a small-signal position sinusoid at

20 Hz was observed. The results were that time delay improved with more K_{VFR} . For zero G_{PVFR} , the time delay was decreased by about 40%; for 75% G_{PVFR} , the time delay was reduced to less than half.

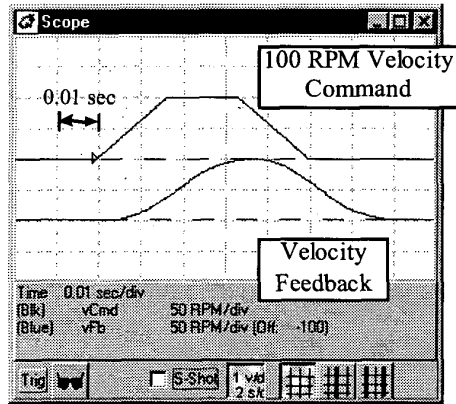


Fig. 11. Trapezoidal move use for stability analysis

D. Settling time

The time required after the deceleration command ends (that is, at the end of the trapezoid) until the velocity is under one RPM was observed. The results were that settling time delay also improved with more K_{VFR} . For zero position FF, the time was decreased by about 1/3; for 75% position FF, the time delay was reduced almost by half.

E. DC following error

DC following error is the distance between the command and feedback when the motor is rotating at constant speed. For any given set of gains, this is proportional to speed and is given in degrees/1000 RPM. This was verified on the working system. The results were that following error also decreased with more K_{VFR} because larger K_{VFR} allowed larger values of G_P . The following error was reduced approximately 30%-40% by raising K_{VFR} from 0% to 100%.

F. Dynamic stiffness

Dynamic stiffness is the ability of the system to maintain speed or position when subjected to a disturbance torque. A disturbance torque is injected into the model as shown in Figs. 3 and 4. The response of velocity is measured, both in the time domain and the frequency domain. Dramatic increases in low-frequency dynamic stiffness were realized by lowering K_{VFR} .

The dynamic stiffness frequency response of Fig. 12 shows a typical dynamic stiffness for a PDF ($K_{VFR} = 0$) and PI ($K_{VFR} = 1$) system. (Note that this plot shows velocity vs. torque rather than position vs. torque as in Fig. 5. Dynamic stiffness can be plotted either way; the plots are equivalent, albeit rotated forms. The higher the dynamic stiffness, the better. The units are Nm/(rad/sec).

Note that the PDF system provides much better stiffness at low speeds, but at the worst case frequency (29.2 Hz) the PI system is slightly stiffer. At high frequencies, both controllers are equivalent; that is because for frequencies well

above the bandwidth of the velocity controller, the primary resistance to disturbance is the total system inertia [8]. The data recorded in Tables 1 and 2 are the low frequency dynamic stiffness (2 Hz) and the worst case frequency (29.2 Hz for all gain sets.)

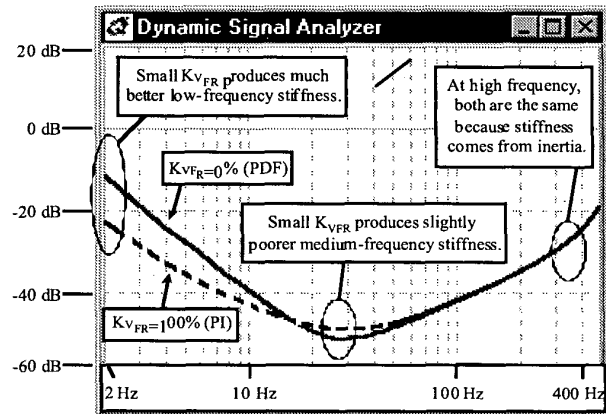


Fig. 12. Dynamic stiffness of PI and PDF controllers

The results were that low-frequency torque disturbance improved with less K_{VFR} . Zero K_{VFR} had 12 dB (4 times) less response to low-frequency disturbance. The worst-case frequency was 29.2 Hz for all tuning parameters; the peak response varied a small amount (< 2 dB.)

G. Step torque disturbance

The final data was collected were the response of the system to a step of 1 Newton-meter. When the system is disturbed with a step torque, the motor shaft moves away from the commanded position. The integrator then ramps up to return the motor to home position. For this test, the rotational distance the motor moved was recorded as well as the time it took to return to within 1° of the commanded position. The less the motor moved and the faster it returned, the better.

The results were that step-torque response improved with less K_{VFR} . For zero G_{PVFR} , reducing K_{VFR} to zero reduced motor movement by 1/3 and, in addition, it settled in about 15% of the time. For 75% G_{PVFR} , the amount of motor movement was reduced more, to little more than half while the settling time was reduced to a little less than half.

VI. CONCLUSIONS

The properly configured state variable ("PID position") motion controller offers complete separation between the command response and the disturbance response tuning. However, two constraints must be met 1) a full command vector must be used, i.e., position, velocity, and acceleration commands must be provided and 2) a feasible command trajectory must be used. Since current industrial servo drives often do not offer acceleration command inputs, this separation property is lost, even for feasible motion trajectories such as trapezoidal velocity profiles.

For cases when no acceleration command is provided or available and a classical cascaded motion control structure is used, the PI+ inner velocity loop controller can be quite attractive when compared to either PI or PDF controllers. However, guidelines to handle the cross-coupling of the tuning must be understood. Under these conditions:

- All command measures improved with more K_{VFR} (more like PI), and almost all disturbance rejection measures worsened.
- Using G_{PVFR} and K_{VFR} , the user can move to the full extremes of maximized command response (G_{PVFR} and K_{VFR} high) and maximum dynamic stiffness (G_{PVFR} and K_{VI} low).
- The PI+ controller is more flexible than either PI or PDF.
- Applications requiring the highest command response should use higher G_{PVFR} and K_{VFR} .
- Applications requiring the highest level of low-frequency disturbance rejection should reduce G_{PVFR} and K_{VFR} .
- For most applications, the selection will be in the center.
- Use of the PI+ method rather than just PI or PDF allows the widest range of choices for system designers.

VII. ACKNOWLEDGMENT

The authors wish to acknowledge the support and motivation provided by Kollmorgen Corporation and by the Wisconsin Electric Machines and Power Electronics Consortium (WEMPEC) of the University of Wisconsin-Madison.

VIII. REFERENCES

- [1] Phelan, R.M., *Automatic Control Systems*, Cornell University Press, 1977.
- [2] Biernson, G., *Principles of Feedback Control, Volume I*, Wiley, 1988.
- [3] G. A. Perdikaris, "A Microprocessor Algorithm for Digital Servo Loops," Conference on Applied Motion Control, Minneapolis, June 1985.
- [4] D. Y. Ohm, "A PDF Controller for Tracking and Regulation in Motion Control," Proceedings of 18th PCIM Conference, Intelligent Motion, Philadelphia, pp.26-36, October 21-26, 1990.
- [5] D. Y. Ohm, "Analysis of PID and PDF Compensators for Motion Control Systems," IEEE IAS Annual Meeting, pp. 1923-1929, Denver, Oct.2-7, 1994.
- [6] Lorenz, R.D., "Synthesis of State Variable Controllers for Industrial Servo Drives," R. D. Lorenz, *Proc. of the Conf. on Applied Motion Control*, pp. 247-251, June 10-12, 1986.
- [7] Lorenz, R.D. and M.O. Lucas, "Synthesis of State Variable Motion Controller for High Performance Field Oriented Induction Machine Drives," in *Proc. of IEEE, IAS Conf. 1986*, pp. 80-85.
- [8] DeDoncker, R.W., L. J. Garces, F. Profumo, R. D. Lorenz, T. Nondahl, and V. Stefanovic, *Microprocessor Control of Motor Drives and Power Converters*, IEEE Tutorial Course Note Book from 1991, '92, & '93 IEEE, IAS Annual Meetings, IEEE Publishing Services # THO587-6.
- [9] Lorenz, R.D. and K. VanPatten, "High Resolution Velocity Estimation", in *IEEE Trans on Industry Applications*, Vol. 27, No. 4, July/August, 1991, pp. 701-708.
- [10] Brown, H., S. Schneider, and M. Mulligan, "Analysis of Algorithms for Velocity Estimation from Discrete Position Versus Time Data," *IEEE Trans. on Ind. Electronics*, Vol. 39, No. 1, Feb 1992.
- [11] Lorenz, R.D., T.A. Lipo and D.W. Novotny, "Motion Control with Induction Motors," in *IEEE Proceedings Special Issue on Power Electronics and Motion Control*, August, 1994, pp. 1215-1240.
- [12] Lorenz, R.D. "Advances in Electric Drive Control", in *Proc. of IEMDC Conf.*, May 10-12, 1999, Seattle, pp. 10-19.
- [13] Schmidt, P.B. and R.D. Lorenz, "Design Principles and Implementation of Acceleration Feedback to Improve Performance of DC Drives", in *IEEE Trans. on Ind. Appl.*, May/June 1992, pp. 594-599.
- [14] Vukosavic, S., and M. Stojic, "Suppression of Torsional Oscillations in a High-Performance Speed Servo Drive," *IEEE Transactions on Industrial Electronics*, Vol 45, No 1, Feb 1998.

TABLE 1: COMMAND AND DISTURBANCE RESPONSE, LOW G_{PVFR} .

Type	PDF						PI
G_P	372	414	475	503	548	588	
G_{PVFR}	0	0	0	0	0	0	
K_V	3000	3000	3000	3000	3000	3000	
K_{VI}	9835	8600	6950	6040	4420	1520	
K_{VFR}	0 (0%)	250 (25%)	500 (50%)	600 (60%)	750 (75%)	1000 (100%)	
Resp. to sin	BW	20 Hz	20 Hz	22 Hz	23 Hz	24 Hz	26 Hz
	Lag	-119°	-107°	-101°	-90	-95°	-91°
Trap	Delay	16.4	14.6	12.7	12.2	11.1	9.9 ms
	20 Hz	ms	ms	ms	ms	ms	
Resp.	Settle time	26.0	24.4	22.1	20.6	19.1	17.1 ms
to DC	Error/krpm	97°	87°	76°	72°	66°	61°
Dist resp.	At 2 Hz	-12.5	-12.8	-13.4	-14.1	-16.0	-23.6
		dB	dB	dB	dB	dB	dB
to sine torque	At 29 Hz	-51.3	-51.2	-51.0	-50.8	-50.4	-49.1
		dB	dB	dB	dB	dB	dB
Pos. resp.	Max error	17°	17°	18°	18°	19°	22°
to step torque	Settle time	33 ms	33 ms	36 ms	39 ms	52 ms	158 ms

TABLE 2: COMMAND AND DISTURBANCE RESPONSE, HIGH G_{PVFR} .

Type	PDF						PI
G_P	148	181	225	244	270	256	
G_{PVFR}	75%	75%	75%	75%	75%	75%	
K_V	3000	3000	3000	3000	3000	3000	
K_{VI}	8140	7625	6500	5720	4225	1240	
K_{VFR}	0 0%	250 25%	500 50%	600 60%	750 75%	1000 100%	
Resp. to sin	BW	26 Hz	27 Hz	30 Hz	32 Hz	36 Hz	44 Hz
	Lag	-93°	-78°	-69°	-67°	-66°	-66°
Trap	Delay	10.0	8.1 ms	6.6	6.1	5.5	4.6
	20 Hz	ms	ms	ms	ms	ms	ms
Resp.	Settle time	17.1	15.6	13.1	12.2	11.0	9.1 ms
to DC	Error/krpm	61°	50°	40°	36°	32°	36°
Dist resp.	At 2 Hz	-12.5	-12.8	-13.4	-14.1	-16.0	-23.6
		dB	dB	dB	dB	dB	dB
to sine torque	At 29 Hz	-49.4	-49.5	-49.6	-49.6	-49.4	-48.5
		dB	dB	dB	dB	dB	dB
Pos. resp.	Max error	22°	21°	22°	23°	25°	37°
to step torque	Settle time	114 ms	98 ms	86 ms	85 ms	92 ms	251 ms