VOL. 3, NO. 3, JUNE 2008 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

DESIGN AND DEVELOPMENT OF A ROBUST CONTROL ADJUSTABLE ELECTRICAL DC DRIVE SYSTEM USING PI CONTROLLER

Liaquat Ali Khan, Abrar Ahmed, Umar Abdul Ahad and Syed Zahid Hussain
Department of Mechatronics Engineering, Air University, Sector E-9, Islamabad, Pakistan
E-Mail: kliaquat@yahoo.com

ABSTRACT

Electrical drives lie at the heart of most industrial processes and make a major contribution to the comfort and high quality products we all take for granted. Electrical drives involving different types of electrical motors turn the wheels of industry. In an industrialized country, more than 60% of the generated electrical energy is used in motor drives. The application of electrical drives spread from low fractional horse power applications in instruments to the industrial applications. Wide power, torque and speed ranges, adaptability to almost every operating condition, high efficiency, fast response, control simplicity, ability to operate as a generator in braking mode and various mechanical design types make the electrical drive very competitive among the other drive types. This work is based on the Robust Control Adjustable Electrical DC Drive System using PI (Proportional Integrator) controller. It encompasses the development of the DC drive. It also includes the design and fabrication of the mechanical load wheel structure. Thus the work finally gave a product in the form of a test jig for checking the wear and tear of a small metallic material after being spring pressed and scrubbed on the edged copper face of the aluminum disk wheel. The integrity of the system is based on keeping the wheel speed constant. In nullifying the steady state error the PI control algorithm was eventually used with root locus design method that could enable finding the PI coefficients. It turns out to be a robust and resilient drive that keeps the load wheel speed invariable at disturbances. The theoretical model is validated with the experimental results.

Keywords: DC drives, PI controller, steady state error, root locus, robust control.

INTRODUCTION

There are numerous methods adopted on the control philosophy of speed drive. Such as the one found in the paper written by Kadwane [1], titled "Converter Based DC Motor Speed Control Using TMS320LF2407A DSK", deals with real time DC motor speed control, using the low-cost new generation TMS320LF2407A digital signal processor. A PID controller is designed using MATLAB functions to generate a set of coefficients associated with the desired controller characteristics. The controller coefficients are then discretized and included in an assembly language or C program that implements the PID controller. The technique adopted in our work is the implementation of PI algorithm that controls the phase angle of the pulsating DC derived from the mains and sent to the load. The converter is the analog circuitry that makes ramp from the mains and compares with the analog voltages of DAC output. Thus the moment the ramp crosses the analog level, the gate firing is activated. DAC derives its analog voltages from microcontroller 8-bit value places at its input. Thus microcontroller derives percent of power in the form of 8-bit from PI algorithm after taking error of set speed and run speed from digital speed encoder.

Design Specifications

Atmel® 89C52, 8-bit Microcontroller was used as an on-board computer in the development of drive. The Proportional-Integral Control algorithm was explicitly used in the assembly language source code embedded in the Microcontroller. Following is the work model:

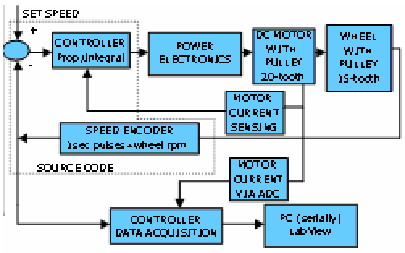

Figure-1. Working model of a DC motor.

The drive is quadrant-I and Non Regenerative. Over current and torque protection is included to secure Power Electronics of the drive. Open loop feedback protection is further embedded in the drive. This would save the motor from abnormal run at the malfunction of speed encoder or breaking of motor belt. The drive is capable of controlling the load speed of the DC motor in the range of 1500 to 2400 rpm. Besides, a run time in the range of 1 to 9999min is taken from the user. The drive automatically stops the motor after the elapse of the desired run time. The rpm-time relationship derives the exact revolutions of the load. A DC shunt motor of 60V/5A and 1500 rpm is used. The speed is enhanced to 2400 rpm by belt-pulley mechanism connecting the motor shaft to the load shaft. The load is a 2 kg aluminum disc wheel. The entire outer edge of the disc wheel is copper lined. The copper lining was made to enable scrubbing a job piece at the outer lining of the wheel during its rotation. The job piece may be a carbon brush of 40.5mm length, 12.5mm width and 24.4mm thickness. Thus two separate job pieces at the two opposite outer ends of the wheel are held for scrubbing. The assembly was made in

VOL. 3, NO. 3, JUNE 2008 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

such a way that each job piece is constantly spring pressed at one face and the other face is scrubbed by the outer lining of the wheel. The tightness of spring is variable. Thus two such similar assemblies are made accordingly. During set speed and desired run time the job piece keeps wearing out as a result the spring tightness varies. This will affect the running rpm of the load wheel. Had this not been a robust control the wheel speed would have increased due to looseness of spring tightness. But the robustness of the drive does not let the speed distract. Thus the load wheel speed remains constant throughout the operation of the drive. The crux of this scheme was to analyze the wear and tear in the thickness of the job piece after having scrubbed by the wheel to the known number of revolution at set speed/time.

The Power Electronics of the drive takes analog voltages from DAC (Data Acquisition Card) to vary the pulsating DC voltage by Phase Angle Control that is eventually sent to the motor. DAC is 8-bit and interfaced to one of ports of Microcontroller. Thus the Microcontroller calculates the percent of power needed for the motor via its PI algorithm after deriving the speed error. It is primarily the percent that is output to DAC. Consequently the drive voltages to motor are varied in accordance to the percent of power calculated in microcontroller. Thus at start of run operation the overshoot is not any concern but settling time around 20 sec is desired. But PI algorithm constitutes the integral anti wind up source code that does not let the drive reach over saturation. A maximum overshoot of 30% is tolerable in the transient state. Once the drive achieves the steady state in the desired settling time, the job piece are tightened thus causing the speed to lowered, now it is the PI algorithm that works and brings back the running speed to the desired speed.

There is another exceptional feature embedded in the drive: the drive is interfaced to PC Com Port via LabView Software using a second Microcontroller. Thus the real time behavior of the drive for the measured speed to the set speed from start to the completion of cycle can be viewed graphically. Besides, the current drawn by the motor can also be viewed graphically. The graphical results are saved in real time in Excel File (*.xls) for later behavioral analysis of the drive.

Design details

Rotational mechanical systems are handled the same way as translational mechanical system except that

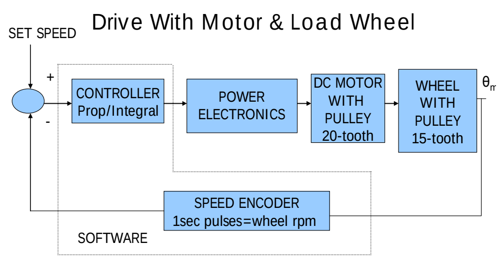

Figure-2. Drive model with mechanical outfits.

torque replaces force and angular displacement replaces translational displacement. The mechanical components moment of inertia, spring constant and damping coefficient for rotational systems are the same as those for translational systems, except that the components undergo rotation instead of translation. Since torque defines the “pushing or rotating force”, here it is the force of the motor. Force can only be measured when pushing against something that pushes back equally hard. If that sounds odd, consider how easy it is to hit a punching bag "hard" and how difficult it is to punch into the air with the same force. The motor-torque specification is really a measurement of the force pushing back against the motor. The torque specification of the motor is for the maximum torque provided by the motor at the nominal voltage. In a DC motor, the motor constants K_m (voltage constant) and K_T (torque constant) are related to geometry factors, such as dimension of the rotor and number of turns in armature winding. In ideal conversion case K_m and K_T are treated identical. Thus the electrical to mechanical and vice versa conversion are identical [1].

MATHEMATICAL MODELLING

In view of the drive model depicted in Figure-2, the transfer function of the DC motor will be developed for a linear approximation to an actual motor, and second order effects, such as hysterises and the voltage drop across the brushes, will be neglected. The linearity of many mechanical and electrical elements can be assumed over a reasonably large range of variables. This is not usually the case for thermal and fluid elements, which are more frequently non linear in character. The motor used in the work is the armature-current-controlled DC motor utilizing a constant field current.

The motor circuitry in Figure-3 implies that,

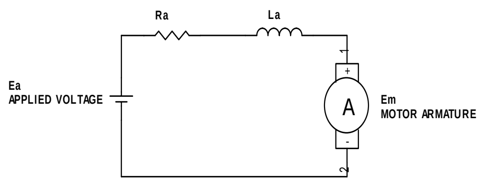

Figure-3. Armature current controlled DC motor.

In this system DC motor is driving a load wheel. The wheel is made of Aluminum with an outer copper lining on the edges. It is 2kg mass. The outer lining of the wheel is spring pressed with varying force. The motor drives the wheel by a pulley called driver pulley. It has 20-tooth. At the wheel there is another pulley attached called driven pulley. It has 15-tooth. Both driver and driven pulleys are linked by hard tightened tooth-belt. The teeth in the pulley act as a gear to provide mechanical advantage to rotational systems. Like going uphill, one shifts to provide more torque and less speed. On the straight way one shifts to obtain more speed and less torque. Thus, gears allow one to match the drive system and the load- a trade-off between speed and torque. For many applications, gears exhibit backlash, which occurs because

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

of the loose fit between two meshed gears. The drive gear rotates through a small angle before making contact with the meshed gear. The result is that the angular rotation of the output gear does not occur until a small angular rotation of the input gear has occurred. It is assumed that in our system, there is no backlash.
System's mechanical outflow is shown in Figure-4.

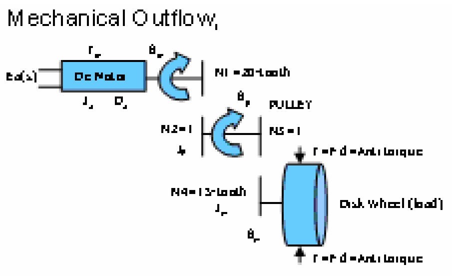

Figure-4. Dynamic model of the system.

It must be noted that $F_d(s)$ causes a change in moment of inertia of wheel. As $F_d(s)$ is varied by making displacement in the pressing of spring, $J_w(s)$ will vary. When $F_d(s)$ is zero, the wheel will move at its original inertia. Therefore in this calculation of finding the transfer function, we assume original moment of inertia of wheel without being disturbed by $F_d(s)$.

The motor and wheel specifications given by the manufacturer are as follows:

Diameter of Aluminum Load Wheel with outer copper lining = 200mm
Mass of Aluminum Load Wheel with outer copper lining and axle = 2kg
J_m = 0.00025 N-m / radian / sec^2
D_m = 0.008 N-m / radian / sec
K_m = 0.025 Volt-sec / radian
J_w = 0.005 N-m / radian / sec^2
R = 0.5Ω

Thus the Ultimate Transfer Function of motor and load wheel is derived to be,

$$\frac{\theta_w(s)}{Ea(s)} = \frac{7.2072}{s[s+1]} \qquad \text{...Eqn-(1)}$$

Multiplying with mechanical efficiency of 0.72, we have the final transfer function of motor and load wheel,

$$\frac{\theta_w(s)}{Ea(s)} = \frac{5.189}{s[s+1]} \qquad \text{...Eqn-(2)}$$

Power amplifier transfer function

The Power Amplifier (PA) used in the drive takes analog value in the span of 0 to 7V from Digital to Analog It converts DC voltage in the span of 0 to 65V. Power amplifier transfer function is 9.285.

The transfer function of PI algorithm is based on Root Locus Method. For the PI controller Kp and Ki parameters are estimated.

$$PI(s) = K_P + \frac{K_I}{s}$$

$$PI(s) = K_P \frac{(s + \frac{K_I}{K_P})}{s} \qquad \text{...Eqn-(3)}$$

Where K_i/ K_p is the zero of the controller with pole at origin. The design criterion is based on 15% overshoot and settling time around 15sec. Since the dominant factor behind the design is the settling time of steady state, therefore the overshoot is regarded secondary in the transient state once the drive is powered on. By root locus the open loop transfer function of motor and load wheel with power amplifier is given as,

$$\frac{\theta w9s)}{Ea(s)} = \frac{48.179K}{s(s+1)} \qquad \text{...Eqn-(4)}$$

Gain is calculated at 15% overshoot and check the closed loop step response of the drive. Since the transfer function of the drive in Eqn-3 is a second order lag system with zero steady state error, therefore the need for PI controller interfaced to the system comes when there is a loading effect experienced on the load wheel of the drive. To compensate the error the given transfer function would not be able to nullify the change in speed. This could then only be possible manually by the user while increasing the power to nullify the error. The auto operation of this scheme is therefore made possible by PI controller.

Root locus of closed loop transfer function and its step response is shown in Figures 4 and 5.

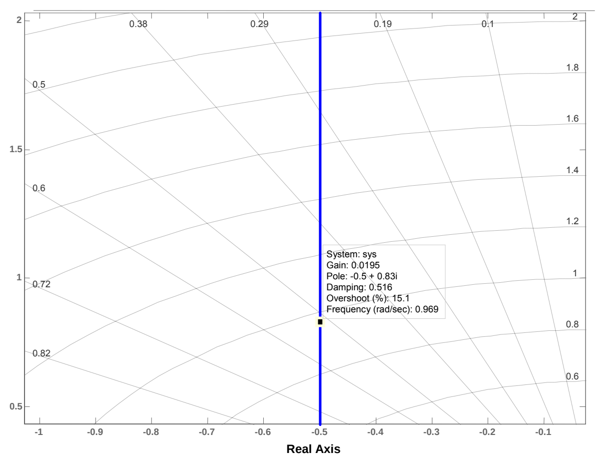

Figure-5. Root Locus without PI Controller.

VOL. 3, NO. 3, JUNE 2008 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

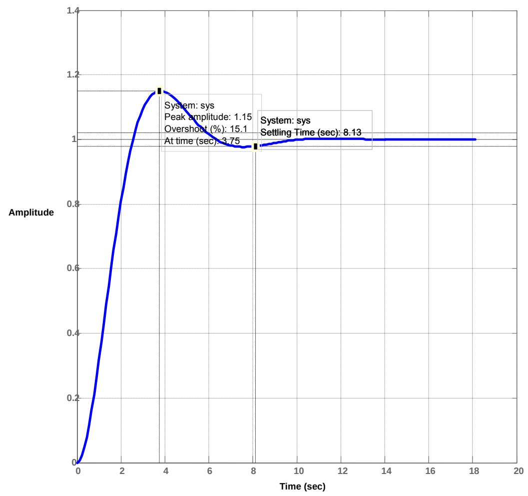

Figure-6. Closed Loop Step Response without PI Controller and spring.

The transfer function of system turns out to be

$$K\frac{s+0.24162}{s}\frac{48.179}{s(s+1)}$$

Or

$$K\frac{(48.179s+11.641)}{s.s(s+1)}$$

...Eqn-(5)

The system drive transfer function now contains PI controller with motor and wheel having gain K yet unknown. It is now to be found for 15% OS using root locus method. The root locus of system drive of Eqn-6 is depicted below in Figure-7.

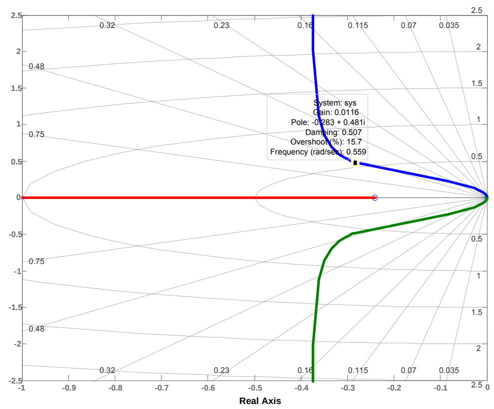

Figure-7. System Root Locus with PI Controller.

The root locus in Figure-7 bears great insight. The root locus has given a gain of 0.0116 at 15.7% OS which can be regarded approximately at 15%. Thus the overall gain of the system becomes 0.558. Therefore the open loop transfer function is found to be

$$T(s)=0.558\frac{(s+0.24162)}{s.s(s+1)}$$ ...Eqn-(6)

Design conclusion at root locus technique

Eqn-6 is the ultimate open loop transfer function of the DC Drive. We followed this locus and found the following damping ratios (ζ), dominant poles and frequency at 15.7% OS. Also K_p is found to be 0.558, K_i as 0.1348 and $\zeta = 0.507$
Dominant Poles = $-0.283 \pm j0.4811$

The step response of the closed loop transfer function of Eqn-6 is given in Figure-8.

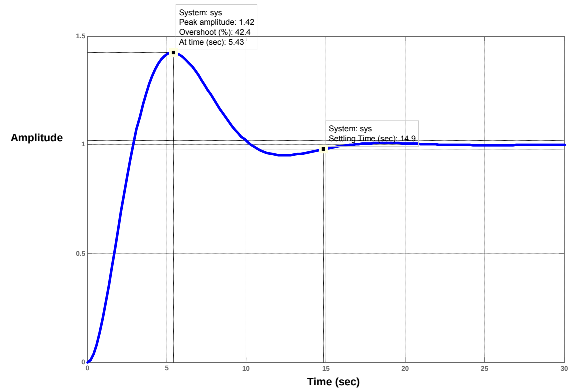

Figure-8. Closed Loop Step Response with PI controller.

From Figure-8, it can be inferred that the overshoot has increased to 42%. It is attributed to PI control or ideal integral controller. As the system is analyzed as second order with dominant poles irrespective of the third pole this overshoot is likely. The exceeding overshoot has occurred since in the design we did not take nonlinearities into account like backlash in motor belt, dead zone in motor and amplifier. Simultaneously we assumed that motor torque is delivered to the load wheel equally irrespective of the fact that it delivered via belt. We did not take the damping of belt mechanism into account. Therefore these nonlinearities add damping into the system which eventuality reduces the overshoots. Simultaneously we are not using derivative control.

Drive real time load speed response

Since the drive is interfaced to computer and the real time load wheel speed and motor current have acquired on LabView®. Step responses in real time are compared with theoretical design. DAC card had made using the pulse width modulation principle.

RESUSTS

Real time step responses at different rpm with disturbances are compared with the theoretical response. Figures 9-12 are showing these responses.

VOL. 3, NO. 3, JUNE 2008 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

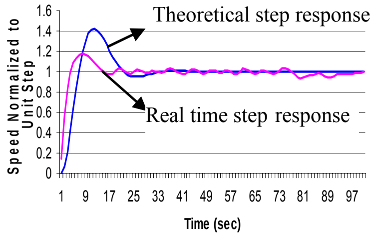

Figure-9. Normalized real time and theoretical response comparison at 2000 rpm.

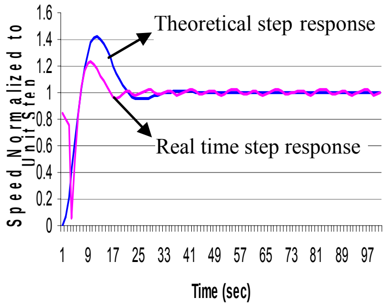

Figure-10. Normalized real time and theoretical response comparison at 1800 rpm.

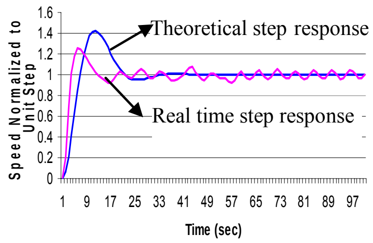

Figure-11. Normalized real time and theoretical response comparison at 1600 rpm.

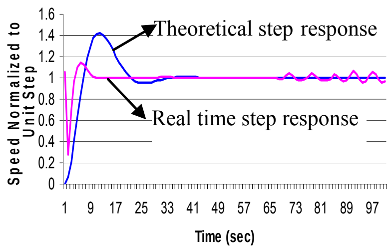

Figure-12. Normalized real time and theoretical response comparison at 1500 rpm.

These comparisons show that the overshoot in theoretical response is higher as compared to the real time results. This is because the damping is not modeled perfectly. Therefore, if one can model the damping perfectly, the graph will exactly match. But it the two responses are very close to each other. Fine tuning of theoretical model can be done using experimental results.

CONCLUSIONS

As the process operation spans around 1500 to 2000 rpm, the average overshoot is observed around 19% with a settling time of 21 sec. It eventually suffices the process requirement and thus the design conclusions for the coefficients of PI proved accurate. The drive can be made controllable at lower rpm provided damping factor is added. This can be made possible if a gear train is added between motor and load wheel; instead of transferring power directly from motor rotor to the load wheel. From Figure-8, the designed overshoot turns out to be 42%. But after analyzing the graphs in Figures 9, 10 and 11 at 2000, 1800 and 1600 rpm the overshoot found as 17.4%, 20.157% and 21.94%. This damping is attributed to belt mechanism and other nonlinearities not taken into account in the drive transfer function. But the settling times are found as 25sec, 22sec and 20sec. These are in match to the designed parameters. Similarly the steady state error is found very small even during disturbance deliberately induced in the drive.

Thus it is inferred that if the system transfer function is developed approximating linearity then using root locus technique one can find the desired pole/zero location for the coefficients of PID algorithm. This drive is proportional integral control only, but one can further add derivative control to investigate the behavior of the load wheel during disturbance. This blank spot of derivative control is left to the reader for further enhancing its operational performance.

NOMENCALTURE

E_a =applied voltage to motor (volts)
T_m =motor torque (N.m)
θ_m =motor angular displacement (radians)
θ_p =pulley angular displacement (radians)
θ_w =load wheel angular displacement (radians)
J_m =moment of inertia of motor (N-m/radian /sec^2)
J_p =moment of inertia of pulley (N-m/radian/sec^2)
J_w =moment of inertia of wheel (N-m/radian/sec^2)
D_m =damping coefficient of motor
F_d =opposing torque on load wheel (N)
K_m =motor constant (Volt-sec / radian)
K_P =proportional constant
K_I =integral constant
R =resistance (ohms)
K =amplifier gain constant
ζ =damping constant

REFERENCES

[1] Norman S. Nise. 2004. Control System Engineering. 4th Ed., John Wiley & Sons.

[2] Kadwane S.G. and Someswara Phani Vepa Karan. 2006. Converter Based DC Motor Speed Control using TMS320LF2407A DSK. 1st IEEE conference on Industrial Electronics and Applications.

ARPN Journal of Engineering and Applied Sciences

[3] Charles L. Phillips and Royce D. Harbor. 2000. Feedback control system. 4th Ed., Prentice Hall Inc.

[4] Richard C. Dorf and Robert H. Bishop. 1998. Modern Control Systems. 8th Ed., Addison-Wesley.

[5] Katsuhiko Ogata. 2002. Modern Control Engineering. 4th Ed., Prentice Hall Inc.