DESIGN PID CONTROLLR USING DYNAMIC PROGRAMING ALGORITHM TECHNIQUES UNDER MATLAB/SIMULINK APPLIED TO A REAL PLANT

[1]ADEL A M ABOSDEL, [2]AHMED JABER ABOUGARIR

[1]Electrical and Computer Engineering, AL Zaytuna University- Libya.
[2]Electrical and Electronic Engineering, Al-Fateh University-Libya
E-mail: adel1880@yahoo.com, abougarir@yahoo.com

I- INTRODUCTION

PID (Proportional-plus-Integral-plus-Derivative) is a popular control method extensively used in an industrial set up. The advantages of a PID controller include its simple structure along with robust performance. One the performance measure for a system has been chosen, the next task is to determine a control function that minimizes this criterion.

Two methodes of a accomplishing the minimization are the minimum principle of Pontryagin and the methode of dynamicprograming.
The desired performance can be readily stated in terms of time domain performance indices, for example, minmum time problem and tracking problem. Therefore the systems that are adjusted to provide a minimum performance index are often called optimal control system.

II- DESCRIPTION OF THE PHYSICAL SYSTEM FOR OPTIMISATION

Application and principles of automatice control and automation is found almost everywhere in everyday life. Disk drive system in figure 1, used in computers of all sizes , they represented an important application of optimal control system .Disk drives system contain a mechanism and a controller. The mechanism is made up of the recording components (the rotating disks and the heads that access them) and the positioning components (an arm assembly that moves the heads into the correct position together with a track-following system that keeps it in place). The disk controller contains a microprocessor, some buffer memory. The controller manages the storage and retrieval of data to and from the mechanism and performs mappings between incoming logical addresses and the physical disk sectors that store the information.

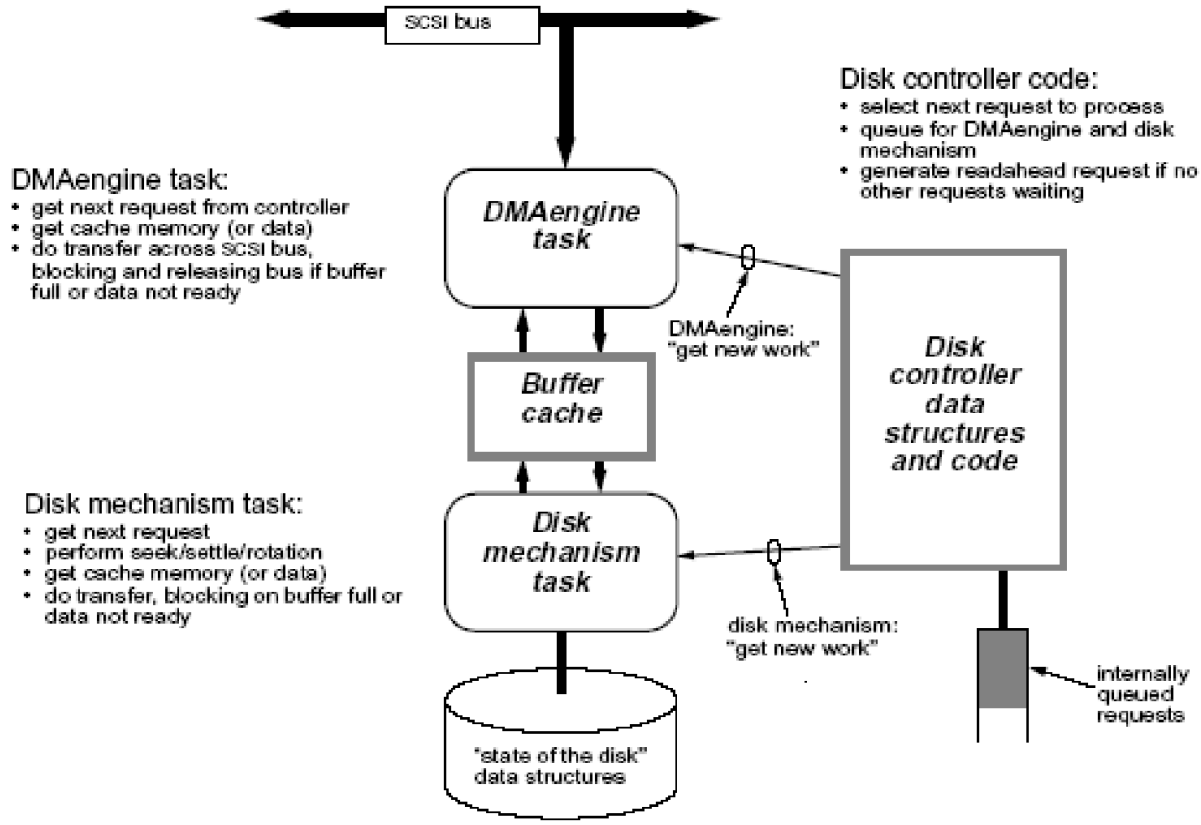


Figure 1:Structure of Disk Drive System

III. CONTROL SYSTEM DESIGN

Control engineering is based on the foundations of feedback theory and linear, non-linear system analysis. A feedback control (figure 2)is a control system that lends to maintain a prescribed relationship of one system variable to another by comparing functions of these variables and using the difference as

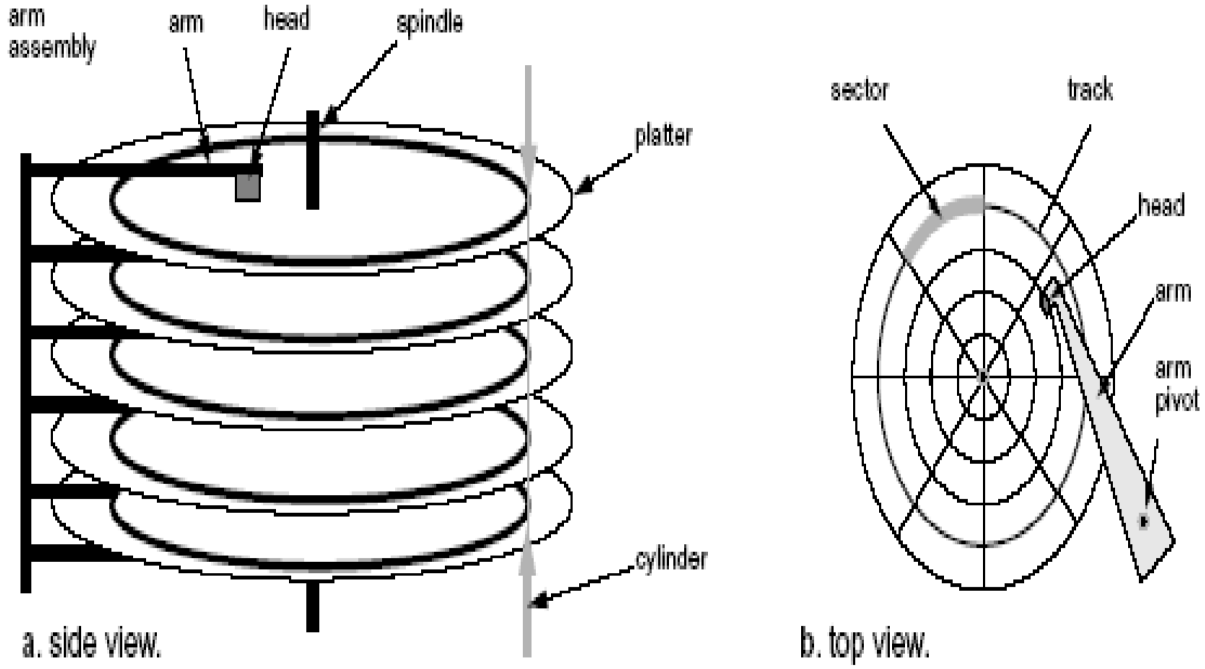


Figure 2 : Block diagram of a feedback control system

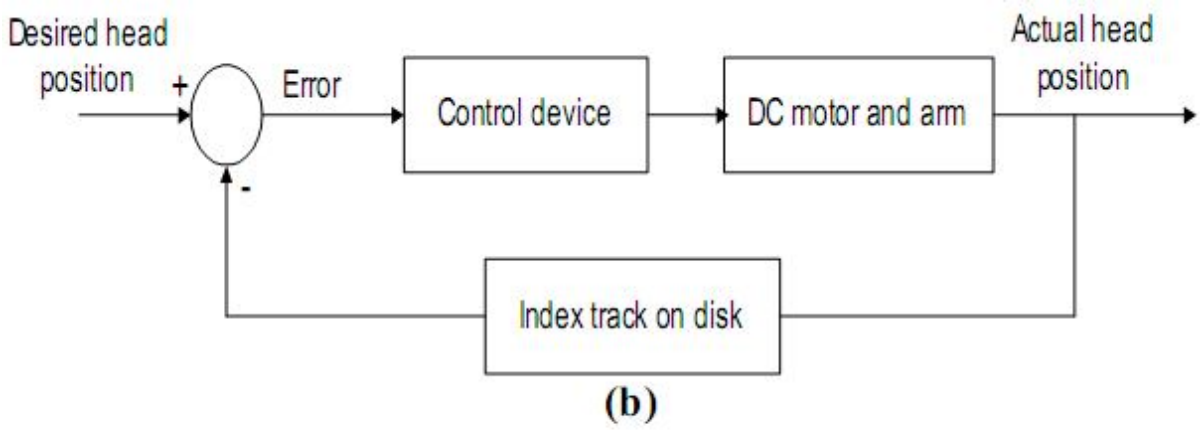


Figure 2 : Block diagram of a feedback control system

IV. MATHEMATICAL MODELS OF DISK DRIVE READ SYSTEM

System modeling results in a mathematical representation of the process under investigation in figure 3. .In engineering, the models are also used for control system design, fault detection and simulation.

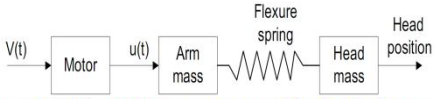


Figure 3: Block diagram of disk drive system

I- Transfer function of DC motor

$V_f(s) = (R_f + L_f S) I_f(s)$

$$I_f(s) = \frac{V_f(s)}{R_f + L_f S}$$

The motor torque is equal to the torque delivered to the load

$T_l(s) = T_m(s) = K_m I_f(s)$,

K_m is defined constant motor

$T_l = jS^2\theta(s) + bS\theta(s)$

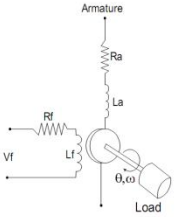
$$= K_m * \frac{V_f(s)}{R_f + L_f S}$$

$$\frac{\theta(s)}{V_f(s)} = \frac{K_m}{s(js+b)(L_f s + R_f)}$$

$$= \frac{K_m / jL_f}{s(s + b/j)(s + R_f / L_f)}$$

$$= \frac{K_m / bR_f}{s(\tau_f s + 1)(\tau_l s + 1)}$$

$\tau_f = L_f / R_f$, $\tau_l = j/b$



Typical parameters for disk drive reader (DC motor)

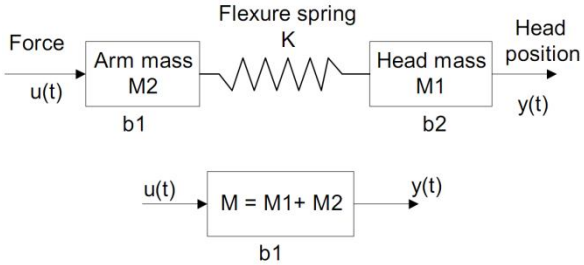
Parameter	Symbol	Typical parameter
Inertia of arm and read head	J	1 N . m . s^2/rad
Friction	b	20 Kg/m/s
Armature resistance	R	1 ohm
Motor constant	K_m	5 N. m/A
Armature inductance	L	1 mH

$$\tau_f = L_f / R_f = 1\,\text{ms} \quad , \quad \tau_l = j/b = 50\,\text{ms}$$

$$\frac{\theta(s)}{V_f(s)} = \frac{K_m / bR_f}{s(\tau_f s+1)(\tau_l s+1)} = \frac{5000}{s(s+20)(s+1000)}$$

$$\frac{\theta^*(s)}{V_f(s)} = \frac{K_m / bR_f}{(\tau_f s+1)(\tau_l s+1)} = \frac{5000}{(s+20)(s+1000)}$$

II- Transfer function of Arm and read head



Here we identify the arm mass as M_1 and the head mount mass M_2. The flexure spring is represented by the spring constant K. The force u(t) to drive the mass M_1 is generated by DC motor , If the spring is absolutely rigid (nonspring) , then we obtain the simplified model ($M=M_1+M_2$) .

$$M\frac{d^2y}{dt^2} + b_1\frac{dy}{dt} = u(t)$$

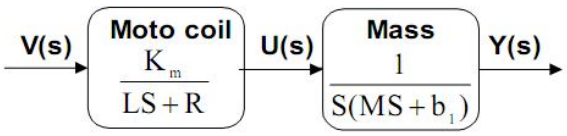
$$\frac{Y(s)}{U(s)} = \frac{1}{S(Ms + b_1)}$$

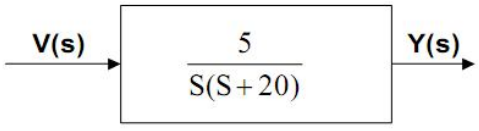
Typical parameters for disk drive reader (Arm and head reader)

Parameter	Symbol	Typical parameter
Arm mass	M_1	20 g
Read mounting	M_2	0.5 g
Friction at mass 1	b_1	0.41 kg/m/s
Friction at mass 2	b_2	0.0041 kg/m/s

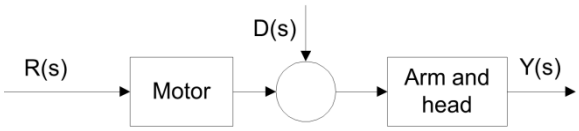
$$M = M_1 + M_2 = 20.5\ \text{g}$$

$$G_L = \frac{Y(s)}{U(s)} = \frac{1}{S(0.0205s + 0.410)}$$

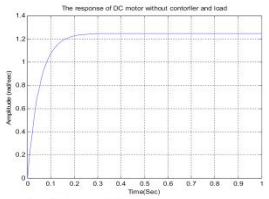




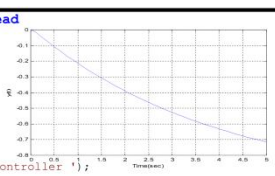
V. RESPONSE THE DISK DRIVE READ SYSTEM WITHOUT CONTROLLER



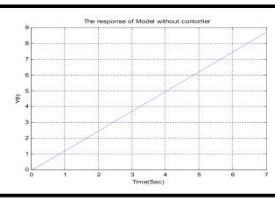

```
% find the speed motor with out controlled
% transfer function of system is
% G(s) = km/((js+b)(ls+r));
km=5;
b=20;
r=1;
l=0.001;
j=1;
u=5;
t=[0 : 0.01 :1 ];
num=[km];
den= [(j*l) ((j*r)+(b*l)) (b*r) ];
y=step(u*num,den,t);
plot(t,y),grid
xlabel('Time(Sec)'),
ylabel('Amplitude (rad/sec)')
title('The response of DC motor without controller and load');
```




```
% Find the position of reader head
% Transfer function for open loop ;
% G(s) = 5/s(s+20)
u=5;
km=5
t=[0 : 0.01 :5 ];
num=[km];
den= [1 20 0];
y=step(u*num,den,t);
plot(t,y),grid ;
xlabel('Time(Sec)'),ylabel('Y(t)');
title('The response of Model without controller ');
```




```
% The Effect of Disturbance
% ( Noise Signal )
% G(s) = 5/s(s+20)
km=5
nc=[km]; dc=[1];
ng=[1];dg=[1 20 0];
[n,d]=feedback(ng,dg,nc,dc);
n=-n;
y =step(5*n,d,t);
plot(t,y),grid
ylabel('y(t)'),xlabel('Time(sec)');
```



VI. PERFORMANCE OF DISK DRIVE READ SYSTEM

This case study demonstrates the ability to perform classical control design by going through the design of a computer disk drive read/write head position controller. How do you design the controller that meets your performance needs, it's reduced the effect noise signal and improveing the stability. Noise signal preduced from the fan of microprocessor of computer and from power supply . (The optimal solution filter minimizes a measure of the error between your desired response and the actual controller response.)

Selection performance measure :-

1- **Minimum time problem :-** to transfer reader head from initial state position $y(t_o) = 0$ to specified target (desired position) in minimum time .

t_s : settling time = final time = $t_f \le 400$ ms .

$$J_1 = t_s - t_o = t_f - t_o = \int_{t_o}^{t_f} dt \le 400\ \text{ms} \quad , \quad J_2 \le 200\ \text{ms}$$

2 - Tracking problem :- to maintain the system state position y(t) as close as possible to the desired state [r(t) = 5 ' number of trace on disk ' (desired position)] in the interval , [t_r , t_f] . where (t_r is rise time) .

$$J_2 = \int_{t_r}^{t_f} \|y(t) - 5\|^2 dt < 5\ \%$$

VII. SPECIFICATIONS OF THE SYSTEM FOR DYNAMIC PROGRAMING ALGORITHEM

$$\textbf{State vector } x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} \text{Position of rader head} \\ \text{Velocity of rader head} \end{bmatrix}$$

control vector u(t) = $\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} = \begin{bmatrix} \xi \\ \omega_n \end{bmatrix} = \begin{bmatrix} \text{damping ratio} \\ \text{underdamped natural frequency} \end{bmatrix}$

Parameter of controller = $\begin{bmatrix} K_p \\ K_d \\ K_i \end{bmatrix}$

VIII. PHYSICAL CONSTRAINS OF CONTROL AND STATES

$0 < u_1 = \zeta \le 1 \quad , \quad 10 \le u_2 = \omega_n \le 20$

$X_1(0) = 0 \quad , \quad X_1(t_f) =$ desired position

$X_2(0) = 0 \quad , \quad X_2(t_f) =$ free

t_f, t_r is specified : $t_f \le 400$ ms , $t_r \le 200$ ms

P.O (over shot) <= %5

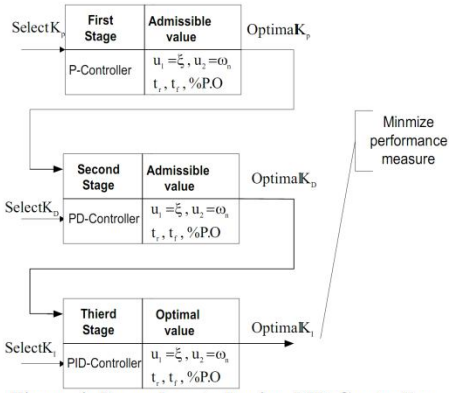
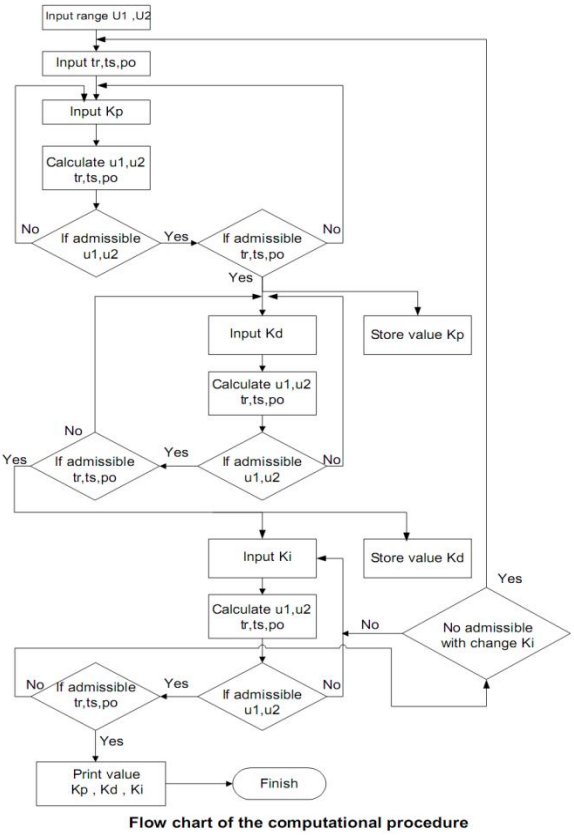


Figure 4: Procedure to Design PID Controller Using Dynamic Programing Algorithm



Flow chart of the computational procedure

Result of the Programme

Please input stage of system ?
3
Please input the physical constrain of control ?
U1= [0 -1] , U2 = [10 - 20]
Please input the range of performance measure?
J1 = minmum time problem <= 400 ms
J2 = rise time <= 200 ms
J3 = tracking problem <= 5%

Costs operation of first stage

K_p	ω	ξ	T_s	%P.O	T_r
0	0.0000	10.000	-	-	22.000
10	7.0702	1.4142	0.4000	-	0.5168
20	10.000	1.000	0.4000	4.3282	0.2765
30	12.2471	0.8161	0.4000	1.1789	0.1929
40	14.1421	0.7071	0.4000	4.3282	0.1542
50	15.8114	0.6325	0.4000	7.7001	0.1243
60	17.3259	0.5774	0.4000	10.858	0.1064
70	18.7011	0.5345	0.4000	13.725	0.0937
80	20.5504	0.5000	0.4000	16.318	0.0840

The optimal value of K_p = 40
The optimal control U1 = 0.7071 *** **The optimal control U2** = 14.1421

Costs operation of second stage

K_d	ω	ξ	T_s	%P.O	T_r
0	14.1421	0.7071	0.4000	4.3281	0.1243
0.25	14.1421	0.7513	0.3765	2.8026	0.1297
0.50	14.1421	0.7954	0.3555	1.6211	0.1351
0.75	14.1421	0.8396	0.3368	0.7788	0.1405
1.00	14.1421	0.8838	0.3200	0.2647	0.1459
1.25	14.1421	0.9280	0.3047	0.0399	0.1513
1.50	14.1421	0.9723	0.2909	0.0002	0.1567
1.75	14.1421	1.0164	0.2783	0.0002	0.1624
2.00	14.1421	1.0606	0.2666	0.0002	0.1675
2.25	14.1421	1.1105	0.2560	0.0002	0.1729
2.50	14.1421	1.1490	0.2461	0.0002	0.1783
2.75	14.1421	1.0932	0.2371	0.0002	0.01834

The optimal value of K_d = 1.50

The optimal control U1 = 0.9723*** **The optimal control U2** = 14.1421

Costs operation of third stage

K_i	ω	ξ	T_s	P.O	T_r
0	0.000	0.000	0.000	-	0.6
1	10.00	1.6625	0.2406	-	0.4191
2	20.00	0.8315	0.2406	0.9137	0.1198
3	30.00	0.5547	0.2406	12.3634	0.0594
4	40.00	0.4156	0.2406	23.8145	0.0375
5	50.00	0.3325	0.2406	33.0537	0.0265
6	60.00	0.2779	0.2406	40.4339	0.0199
7	70.00	0.2375	0.2406	40.4072	0.0159
8	80.00	0.2078	0.2406	51.3196	0.0136
9	90.00	0.1847	0.2406	55.42	0.0111

The optimal value of K_i = 2.00

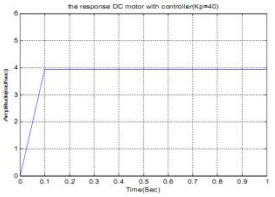
The optimal control U1 = 0.8315 *** **The optimal control U2** = 20

The optimal parameters of PID controller using dynamic programming:-

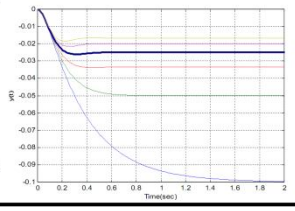
Type of controller	T_s	% P.O	T_r	J
K_p =40	0.4000	4.3281	0.150	0.5900
K_d=1.5	0.2911	0.0002	0.1911	0.4824
K_i=2	0.2406	0.4141	0.120	0.3646

First Stage of Dynamic Programing (PID controller)

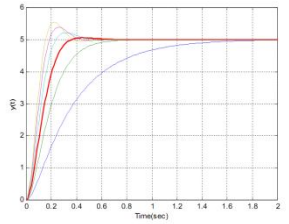
```
km=5; u=5; b=20;
 r=1; l=0.001; j=1;
kp=40; num1=[km];
den1= [(j*l) ((j*r)+(b*l)) (b*r)];
num2= [kp];
den2=[1 ];
 [num3,den3]= series(num1,den1,num2,den2);
[numc,denc]=cloop(num3,den3);
t=0:0.1:1 ;
y=step(u*numc,denc,t);
 plot(t,y),grid
axis([0 1 0 6])
xlabel('Time(Sec)'),
ylabel('Amplitude(rad/sec)')
title('the response DC motor
…… with controller(Kp=40)');
```




```
kp=[ 10 20 30 40 50 60];
t=[0:0.01:2];% for noise signal = 1volt
%u= km*if = km*vf/Rf+slf = km*vf = 5
for i=1:6
nc=[5*kp(i)]; dc=[1];
ng=[1];dg=[1 20 0];
[n,d]=feedback(ng,dg,nc,dc);
n=-n;
[y,x] =step(u*n,d,t);
Ys(:,i)=y
end
plot(t,Ys(:,1),'-',t,Ys(:,2),
……'-',t,Ys(:,3),'-',t,Ys(:,4),'-',
……..t,Ys(:,5),'-',t,Ys(:,6),'-' ),grid
ylabel('y(t)'),xlabel('Time(sec)')
```

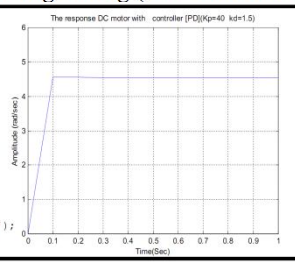



```
% position of read head
kp=[ 10 20 30 40 50 60];
t=[0:0.01:2];
u=5;
for i=1:6
   num=[5*k(i)]; den=[1 20 0];
[n,d]=cloop(num,den,-1);
[y,x] =step(u*n,d,t);
Ys(:,i)=y
end
plot(t,Ys(:,1),'-',t,Ys(:,2),
……'-',t,Ys(:,3),'-',t,Ys(:,4),'-',
……t,Ys(:,5),'-',t,Ys(:,6),'-' ),grid
label('y(t)'),xlabel('Time(sec)')
```

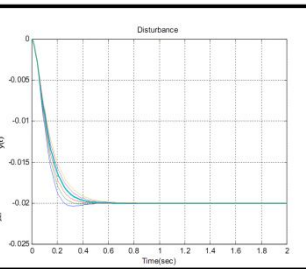


Second Stage of Dynamic Programing (PID controller)

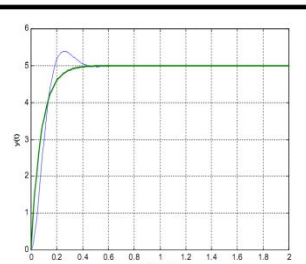
```
% Speed DC motor
kd=1.5;
num1=km;
den1= [(j*l) ((j*r)+(b*l)) (b*r)];
num2= [kd kp];
den2=[0 1];
[num3,den3]= series(num1,
……den1,num2,den2);
[numc,denc]=cloop(num3,den3);
t=0:0.1:1 ;
y=step(u*numc,denc,t);
plot(t,y),grid
axis([0 1 0 6])
xlabel('Time(Sec)'),
ylabel('Amplitude (rad/sec)')
title('The response DC motor
……..with controller [PD](Kp=40  kd=1.5)');
```




```
% Effect disturbance signal
kd=[ 1 1.5 2 2.5 3 3.5];
kp=40;
t=[0:0.01:2];
for i=1:6
nc=[5*k(i) 5*kp]; dc=[ 0  1];
ng=[1];dg=[1 20 0];
[n,d]=feedback(ng,dg,nc,dc);
n=-n;
[y,x] =step(5*n,d,t);
 Ys(:,i)=y
end
plot(t,Ys(:,1),'-',t,Ys(:,2),'-',
……t,Ys(:,3),'-',t,Ys(:,4),'-',
……..t,Ys(:,5),'-',t,Ys(:,6),'-' ),grid
title('Disturbance ');
ylabel('y(t)'),xlabel('Time(sec)')
```

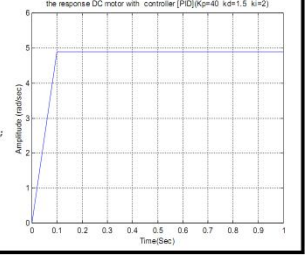



```
% Position of reader head
kd=[ 0   1.5 ];
kp=40
t=[0:0.01:2];
for i=1:2
num=[5*k(i) 5*ka];
den=[1 20 0];
[n,d]=cloop(num,den);
[y,x] =step(5*n,d,t);
Ys(:,i)=y
end
plot(t,Ys(:,1),'-',
……t,Ys(:,2),'-'),
grid
ylabel('y(t)'),
xlabel('Time(sec)')
```

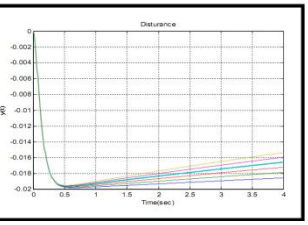


Third Stage of Dynamic Programing (PID controller)

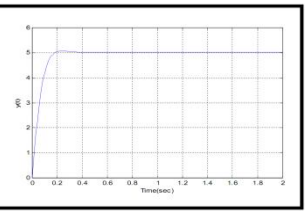
```
% Speed of DC motor under third
% stage of dynamic programing
m=5; u=5; b=20;
r=1; l=0.001;
j=1;
kp=40 ;
kd=2.5;
ki=2;
num1=km;
den1= [(j*l) ((j*r)+(b*l)) (b*r)];
num2= [kd kp ki];
den2=[0 1 0];
[num3,den3]= series(num1,den1,num2,den2);
[numc,denc]=cloop(num3,den3);
t=0:0.1:1 ;
y=step(u*numc,denc,t);
plot(t,y),grid
axis([0 1 0 6])
xlabel('Time(Sec)'),
ylabel('Amplitude (rad/sec)')
title('the response DC motor with
… controller [PID](Kp=40 kd=1.5 ki=2);
```




```
% error signal
ki=[ 1   1.5   2   2.5   3   3.5];
kp=40;
kd=2.5;
t=[0:0.01:4];
for i=1:6
nc=[5*kd 5*kp 5*k(i) ]; dc=[0 1   0];
ng=[1];dg=[1 20 0];
[n,d]=feedback(ng,dg,nc,dc);
n=-n ,[y,x] =step(5*n,d,t);
Ys(:,i)=y , end
plot(t,Ys(:,1),'-',t,Ys(:,2),'-'
……,t,Ys(:,3),'-',t,Ys(:,4),'-',
…t,Ys(:,5),'-',t,Ys(:,6),'-' ),grid
ylabel('y(t)'),xlabel('Time(sec)');
title('Disturbance');
```

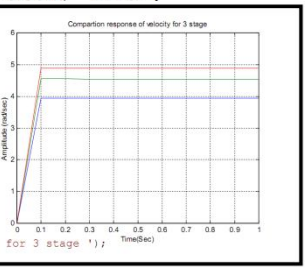



```
% Response of disk drive system under
% thired stage of dynamic programing.
ki=2 ;
kd=2.5;
kp=40
t=[0:0.01:2];
num=[5*kd 5*kp 5*ki];
den=[1 20  0 0];
[n,d]=cloop(num,den);
y =step(5*n,d,t);
plot(t,y),
grid
ylabel('y(t)'),
xlabel('Time(sec)')
```

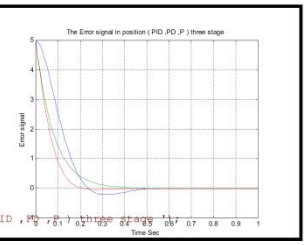


Compare Between Three Stages OF Dynamic Programing (Position - Velocity - Error)

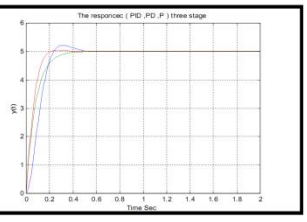
```
% Speed motor for 3 Stage
km=5; u=5;b=20;
r=1; l=0.001;j=1;
kp=40 ; kd=2.5;ki=2;
num1=[75];
den1= [0.001 1.02 95];
num2= [12.5 200];
den2=[0.001 13.52 220];
num3=[12.5 850 10 ];
den3=[0.001 13.52 870 10 ];
t=0:0.1:1 ;
[y1,x,t]=step(u*num1,den1,t);
[y2,x,t]=step(u*num2,den2,t);
[y3,x,t]=step(u*num3,den3,t);
plot(t,y1,t,y2,t,y3),grid
axis([0 1 0 6])
xlabel('Time(Sec)'),
ylabel('Amplitude (rad/sec)')
title(' Compartion response of velocity for 3 stage ');
```




```
kp= 40 ;u = 5 ; kd=2.5;ki=2;
num1=[5*40] ;
den1=[1 20 5*40 ];
num2 = [5*kd 5*kp ];
den2= [1 (20+5*kd) 5*kp];
kp1=70;
num3=[5*kd 5*kp1 5*ki];
den3=[1 ( 20+5*kd) 5*kp1 5*ki];
t=0:0.01:1 ;
[y1,x,t]= step(u*num1,den1,t);
[y2,x,t]= step(u*num2,den2,t);
[y3,x,t]= step(u*num3,den3,t);
E1=u-y1 ,E2=u-y2 ,,E3=u-y3 ;
plot(t,E1,t,E2,t,E3),grid ;
xlabel('Time Sec ');
ylabel('Error signal');
title('The Error signal in position ( PID ,PD ,P ) three stage ');
```




```
kp= 40 ;u = 5 ;kd=2.5;  ki=2;
num1=[5*40] ;    den1=[1 20 5*40 ];
num2 = [5*kd 5*kp ];
 den2= [1 (20+5*kd) 5*kp];
num3=[5*kd 5*kp1 5*ki];
 den3=[1 ( 20+5*kd) 5*kp1 5*ki];
t=0:0.01:2 ;
[y1,x,t]= step(u*num1,den1,t);
[y2,x,t]= step(u*num2,den2,t);
[y3,x,t]= step(u*num3,den3,t);
plot(t,y1,t,y2,t,y3),grid ;
xlabel('Time Sec ');
ylabel('y(t)');
title('The response(Positi
……on)(PID,PD,P) three stage ');
```



CONCLUSION

The designer seeks to satisfy all performance specifications by means of trial-and-error

repetition or adaptation the parameters of controller on- line. The ingenuity and know-how of the designer play an important role in a successful design. In constrained optimization, the general aim is to transform the problem into an easier subproblem that can then be solved and used as the basis of an iterative process. In this way the constrained problem is solved using a sequence of dynamic programing optimization, which in the limit (of the sequence) converge to the constrained problem. this method are now considered relatively inefficient and have been replaced by trial and error or use zigular methode to tune parameters of PID controller.

REFERENCES

[1] B. C. Kou (1995): Automatic Control System, Seventh Edition.
[2] Norman S. Nise (2000): Control System Engineering, Third Edition.
[3] UC Berkeley, A. Packard (spring 2004): Jacobian Linearizations, Equilibrium Points, Lecture Notes.
[4] Prof. Donald E. Kirk: Optimal Control Theory, Naval Postgraduate School, Monterey, California, USA.
[5] Prof. Uy-Loi Ly (2005), Design of Automatic Control Systems, Department of Aeronautics and Astronautics, University of Washington, USA.
[6] The Math Works Inc. (2004), Control System Toolbox Users Guide.

★★★