Optimization of PID Coefficients in an Active Brake System Using MATLAB/Simulink

Mert Bıyık, Emir Ayar

***Abstract*—This paper presents a methodology for optimizing PID controller parameters in an active braking system modeled in Simulink. The system comprises two DC motors, where one motor rotates freely and the other applies a braking force via a PID-controlled torque input. The optimization uses the MATLAB-based fminsearchbnd function to minimize the velocity error between the motors. Simulation results highlight the improvement in performance due to the optimized PID gains.**

***Index Terms*—PID Control, Simulink, MATLAB, Optimization, Active Brake System, DC Motor Control**

I. Introduction

Active braking systems are increasingly important in both automotive and industrial applications where dynamic control of rotational systems is necessary. This project focuses on a simplified but insightful model involving two DC motors in direct mechanical contact. One of the motors is driven by a constant input to simulate a freely rotating wheel, while the second motor operates under a PID control scheme to apply a counteracting torque and simulate braking.

II. Mathematical Model

A. DC Motor Dynamics

The simplified model assumes inertia and back-EMF parameters tuned in Simulink blocks. The block diagram is shown in Fig. 1.

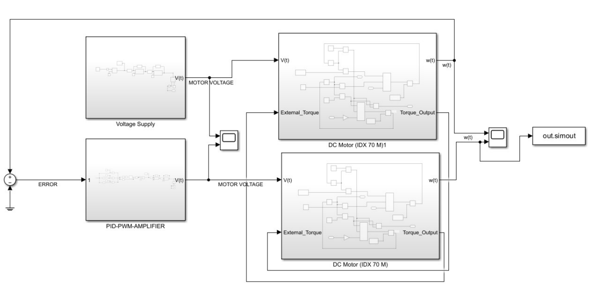

Fig. 1: Simulink model of the Active Brake System

B. PID Control Equation

The PID equation is:

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt}$$

with $K_d = 0$ in this system.

III. MATLAB Cost Function

The cost function calculates the integral of the absolute velocity error:

```
function J = costFunction(x)
    Kp = max(0,x(1));
    Ki = max(0,x(2));
    Kd = 0;
    assignin('base','Kp',Kp);
    assignin('base','Ki',Ki);
    assignin('base','Kd',Kd);
    try
        simOut = sim('ActiveBrakeSystem','StopTime
        y = simOut.simout.Data;
        t = simOut.simout.Time;
        error = abs(y);
        J = trapz(t,error);
    catch
        J = 1e6;
    end
end
```

IV. Simulation Results

A. PID System Output

The velocity response of the braking motor is shown in Fig. 2.

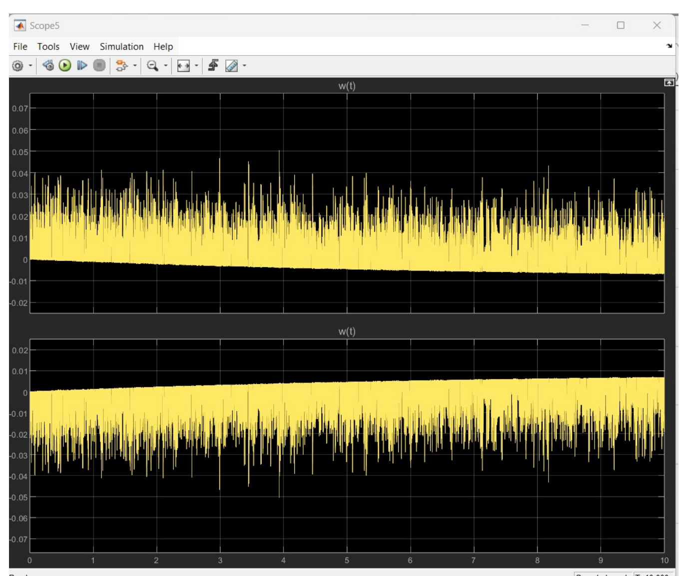

Fig. 2: Motor velocity output (w(t))

B. Control Signal

Fig. 3 illustrates the control voltage signal.

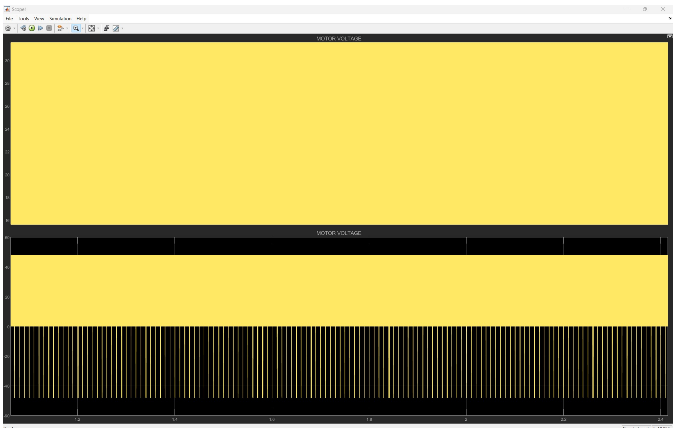

Fig. 3: Control signal (Motor voltage)

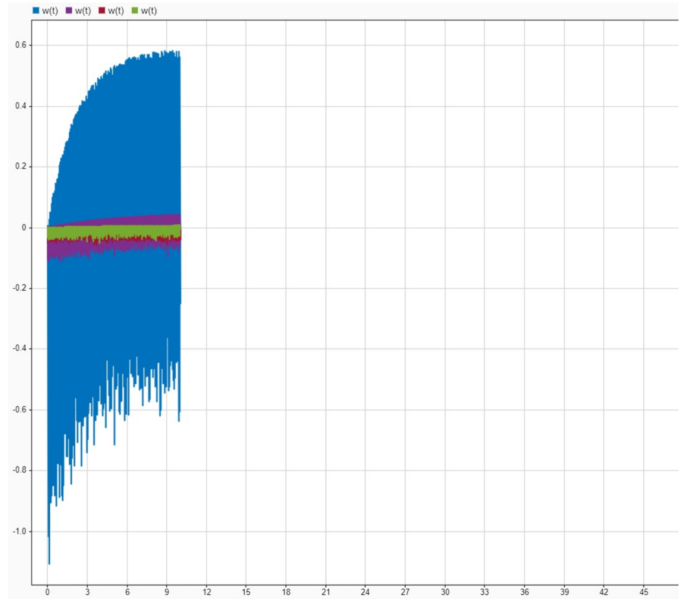

Fig. 5: Beginning, midpoint, and final iterations

C. Optimization Iteration Plots

Initial iteration behavior is shown in Fig. 4.

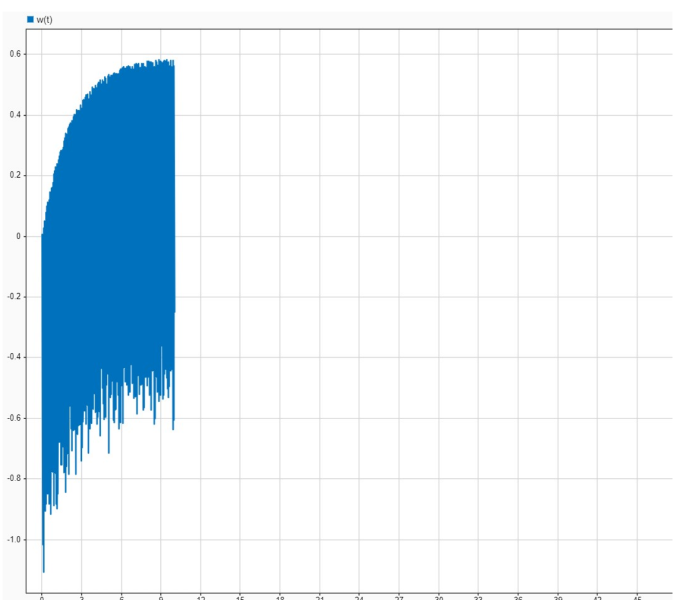

Fig. 4: Output of the first iteration

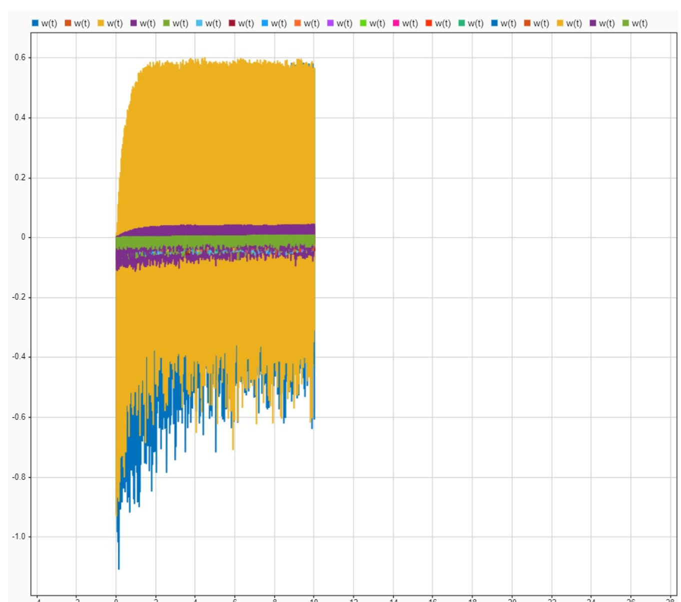

Fig. 6: All iterations overlaid

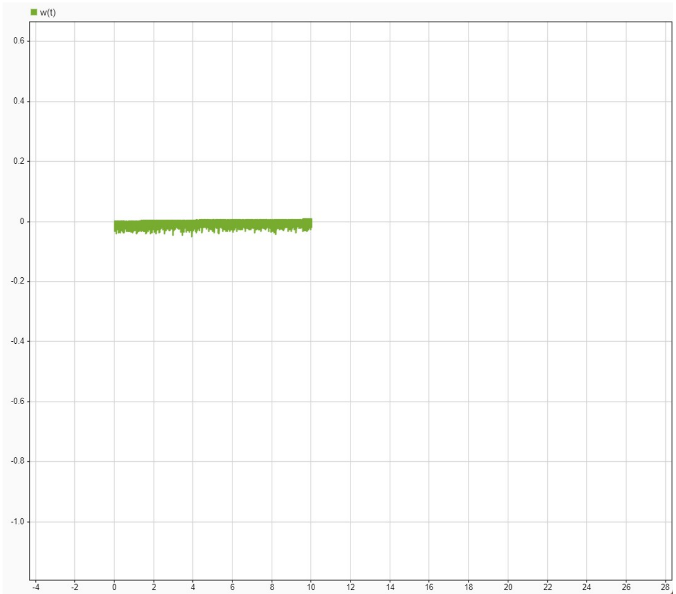

Fig. 7: Final optimized response

V. Conclusion

Simulation and optimization of the braking system controller using fminsearchbnd provided significant improvement in performance. Future work may include physical testing or real-time implementation.

References

[1] J. D'Errico, "fminsearchbnd", MATLAB Central File Exchange.
[2] MATLAB Documentation - PID Controller.
[3] ENM424 Course Notes.