Experimental implementation of MRAC for a Dc Servo Motor

Chaouki T. Abdallah
A. Cerda
R. Jordan

Follow this and additional works at: https://digitalrepository.unm.edu/ece_fsp

Recommended Citation

This Article is brought to you for free and open access by the Engineering Publications at UNM Digital Repository. It has been accepted for inclusion in Electrical & Computer Engineering Faculty Publications by an authorized administrator of UNM Digital Repository. For more information, please contact disc@unm.edu.
EXPERIMENTAL IMPLEMENTATION OF MRAC FOR A DC SERVO MOTOR

A. Cerda, C. Abdallah, and R. Jordan
EECE Department, University of New Mexico
Albuquerque, NM 87131, USA

ABSTRACT
This paper describes a hardware implementation of adaptive controllers using a Digital Signal Processor (TMS320C25). The research explores model reference adaptive control schemes and some of the practical problems associated with their implementation.

1 Introduction
Our research explores model reference adaptive control (MRAC) schemes in a real system. The motivation being that although the theory of adaptive control is highly developed, its practice has been limited [3]. We successfully implemented two MRAC schemes to control a DC servo motor using a digital signal processor. In addition, we studied the following practical aspects of adaptive control:

1. The existence of "Limit Cycles" in the adaptive gains.
2. The influence of the $\gamma$ modification.
3. The influence of the external input.

Other effects were also studied as described in [1] but are not included in this short paper. Additionally, we derived and implemented a new controller that adjusts the $\gamma$ in the application of the $\gamma$ modification [5].

2 Hardware Implementation
The overall system - plant, controller, and display block - is shown in Figure 1. The plant is a DC servo motor. The controller includes the hardware (A/D, DSP board, D/A, DC power amplifier, etc.) and the software that implements the adaptive controllers. The display block contains the parallel port, D/A, oscilloscopes, etc. and the software necessary to display some signals in real time [1].

3 Software Implementation of the controllers
The TMS320C25 is a fixed-point microprocessor so that in our experiments filters, difference equations, sine wave generation and random sequences are implemented using Q format [2].

In our experiments, the plant is given by $K_p/(s+P_m)$, the desired model is $K_m/(s+P_m)$, the output of the plant is $y_p$, that of the model is $y_m$, the input to the plant is $s_p$, and that of the model is $s_m$. The error is defined as $e_1 = y_p - y_m$. All assumptions for the existence of a stable adaptive controller as described in [3] are satisfied. Equation (1) constitutes the adaptive law to adjust the proportional gain [3] for case 1, where only the high frequency gain is unknown: Equation (2) is the discrete version of this law, obtained using the backward approximation [4], where $h$ is the sampling time.

$$k(t) = -\gamma \cdot \text{sigh}(K_p) \cdot e_1(t) \cdot \frac{e_m(t)}{1 + e_k(1)}$$

$$k(t) = -\gamma \cdot \text{sigh}(K_p) \cdot h \cdot e_1(t) \cdot \frac{e_m(t)}{1 + e_k(1)}$$

Equations (3) and (4) represent the adaptive laws of case 2 where both the high frequency gain and the pole are unknown [3]. Equations (5), (6) are the discrete time implementations.

$$k(t) = -\gamma \cdot \text{sigh}(K_p) \cdot e_1(t) \cdot \frac{e_m(t)}{1 + e_k(1)}$$

$$k(t) = -\gamma \cdot \text{sigh}(K_p) \cdot e_1(t) \cdot \frac{e_m(t)}{1 + e_k(1)}$$

4 Experimental Results
4.1 $K_p$ Unknown case
The first set of experiments were designed using equation (2) to study the effect of $\gamma$. We found that $\gamma$ has a dual function in our experiments; it makes the convergence possible by eliminating the limit-cycle effect due to truncation, and it increases the convergence speed of the gains. Figure 2 shows the convergence of the gain $K_p$ for different $\gamma$. There is an important trade-off in the use of $\gamma$: A large $\gamma$ produces large oscillations of the adaptive parameter(s). Figure 3 shows the oscillations around the final values for 3 different $\gamma$.

The other important study performed using the $K_p$ unknown case examined the influence of the input on the convergence of $k(t)$. Figure 4 shows the convergence of $k(t)$ for 6 different step inputs. A
larger magnitude step is shown to produce a faster convergence of the adaptive parameter.

4.2 $K_p$ and $P_2$ Unknowns Case

In the second case, two adaptive gains were adjusted using equations (5) and (6). The input to the model used in this study is a sinusoidal wave form. The amplitude of this sinusoid was adjusted experimentally to make the convergence of $k_t$ and $\theta_t$ possible. The DC value of the sinusoid was set to avoid an input which crossed the level of 0 Volts in order to avoid the nonlinear region of the motor [1]. One of the models used to test the controller (model 1) was:

$$G_m(s) = \frac{100}{s + 10^6}$$

(7)

During the first second of operation, the controller adjusted a large portion of the parameters ($k_t$ and $\theta_t$). Figure 5 shows 1 second of the trajectory of $k_t$ and $\theta_t$, and Figure 6 shows the first 600 microsec of the trajectories of $\theta_{m0}$, $\theta_{p0}$, and $\theta_t$.

An effect similar to the one obtained by updating only $K_p$ was observed in this case. The speed of convergence is related to the magnitude of $\gamma$, as is the oscillatory behavior of the parameters. Figure 7 provides a close look at the oscillations in one of the parameters after 20 seconds of operation of the controller. The curve $C_2k_1$ corresponds to a $\gamma$ of 50, $C_2k_2$ corresponds to a $\gamma$ of 100, and $C_2k_3$ corresponds to a $\gamma$ of 200. Increasing the convergence speed has one drawback; it creates more oscillations in the gains.

![Figure 6: A Close Look at $k_m$, $\theta_p$, and $\theta_t$](image)

Figure 6: A Close Look at $k_m$, $\theta_p$, and $\theta_t$

![Figure 7: Close Look of $k_t$ Oscillations for $\gamma = 50, 100, 200$](image)

Figure 7: Close Look of $k_t$ Oscillations for $\gamma = 50, 100, 200$

4.3 Adjusting $\gamma$

The $(\gamma)$ modification increases the speed of convergence significantly, but it produces an oscillatory behavior around the final value of the adaptive gains. A filter was designed to adjust $\gamma$ as a function of the output error $e_t$. Figure 8 shows a block diagram of this filter. The first block represents a low-pass filter followed by a rectifier that finds the absolute value of the filter output. This value is multiplied by a constant term.

The new controller runs for 18 seconds using a fixed $\gamma$ as the previous controller did. Then the subroutine that implements the $\gamma$ filter is called every 500 microsec and the process of adjusting the factor $\gamma$ begins. The results obtained by using this controller can be observed.

![Figure 8: Adjustment of $\gamma$ schematic block](image)

Figure 8: Adjustment of $\gamma$ schematic block

in Figure 9 ($k_t[0] = 0.5$, $\theta_t[0] = 0.5$ and initial $\gamma = 200$). The trajectories that the parameters follow in both plots is similar. The gains $k_t$ and $\theta_t$ converge to constant values, and the oscillation in the gains disappear.

![Figure 9: Parameter Trajectories for $k_t[0] = 0.5$ and $\theta_t[0] = 0.5$](image)

Figure 9: Parameter Trajectories for $k_t[0] = 0.5$ and $\theta_t[0] = 0.5$

5 Conclusions

The effect of finite register length in digital signal processing is an important factor to consider when a controller is implemented using a fixed-point microprocessor. In our case, the truncation that occurs when a 32-bit hardware register is stored in a 16-bit memory location, is one of the main factors of the limit cycle present in the convergence of the adaptive gains. The solution to this problem was to introduce a modification of $\gamma$. The $\gamma$ modification which has commonly been used to increase the speed of convergence in computer simulation, actually has a dual function in our experiments; it makes the convergence possible, and it can be used to increase the convergence speed of the gains. A larger $\gamma$ produces a faster convergence of the parameters, but unfortunately, the adaptive gains oscillate around the expected final value.

A special controller was designed and implemented to adjust $\gamma$ as a function of the output error $e_t$. This controller reduces the oscillations of the adaptive gains around the final value.

Finally, the magnitude of a step input was shown to have an important influence on the time of convergence of the adaptive gain.

References


3543