

CAN-based Real Time Adaptive Distributed Control

Ana Antunes, Escola Superior de Tecnologia de Setúbal – Instituto Politécnico de Setúbal,
Fernando Morgado Dias, Escola Superior de Tecnologia – Instituto Politécnico de Setúbal,
Alexandre Manuel Mota, Departamento de Electrónica e Telecomunicações - Universidade de Aveiro

Abstract— In this paper the results of a CAN-based distributed real time adaptive control system of a small dc-motor are presented. The goal was to measure the performance of a pole-placement adaptive control algorithm under several jitter conditions due to MAC.

Introduction

It is well known that the use of a distributed control architecture induces jitter problems in the sampling period definition that leads to the degradation of the control performance [1], [2] and [3]. This degradation is seldom evaluated with real-world, real-time systems and most of the results are obtained by simulation. In this paper experimental results obtained under different jitter conditions are presented.

The System

The block diagram of the used system is presented in figure 1. The system includes 4 nodes (sensor, controller, actuator and load generator), based on CANivete boards [4], a small dc motor, additional signal conditioning and power control boards.

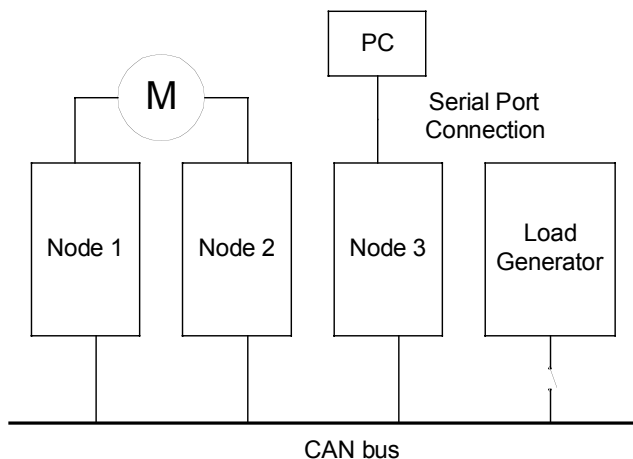


Figure 1 – Block diagram of the system.

The functions of each node vary between different tests as will be seen. Each node

connected to the CAN bus can be sensor and/or actuator and provide the interface with the controller, as needed. The fourth node is used to generate load in the CAN bus.

The controlled system consists of a small 24V dc motor connected to another motor that loads the first one. The load of the controlled system can be changed by changing the resistor connected to the load motor. A block diagram of the system can be seen in figure 2. The controlled variable is the rotation speed of the setup. The speed is measured by an analog tachometer and the

Ana Antunes
Escola Superior de Tecnologia de Setúbal do Instituto Politécnico de Setúbal
Campus do IPS, Estefanilha,
2914-508 Setúbal, Portugal
Tel: +351 265790000
Fax: +351 265721869
Email: aantunes@est.ips.pt

Fernando Morgado Dias
Escola Superior de Tecnologia de Setúbal do Instituto Politécnico de Setúbal
Campus do IPS, Estefanilha,
2914-508 Setúbal, Portugal
Tel: +351 265790000
Fax: +351 265721869
Email: fmdias@est.ips.pt

Alexandre Manuel Mota
Departamento Electrónica e Telecomunicações
Universidade de Aveiro
Tel: +351 234370383
Fax: +351 234381128
Email: alex@det.ua.pt

power applied to the driving (active) motor is controlled by a PWM amplifier. The overall system is very noisy.

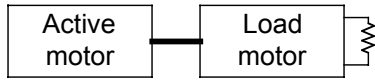


Figure 2 – Block diagram of the controlled system.

Due to the low processing power of the nodes, a PC, connected to the controller node by means of a serial line and running MATLAB, computes the control signal and runs the system identification algorithm.

A picture of the system is shown in figure 3.

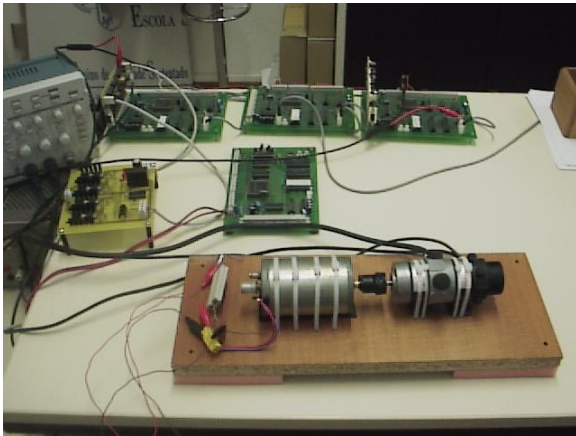


Figure 3 – Picture of the system.

The Adaptive Controller

The system described will be subjected to disturbances due to the load generator that will introduce varying reading and actuation jitter in the system. The controller system used was an adaptive controller. The adaptive controller block diagram is shown in figure 4 [5].

The adaptive controller has two loops. The inner loop includes the variable dynamics controller and the process. The outer loop is composed by the recursive process parameter estimator and a design calculator block and is responsible for the adjustment of the parameters of the controller.

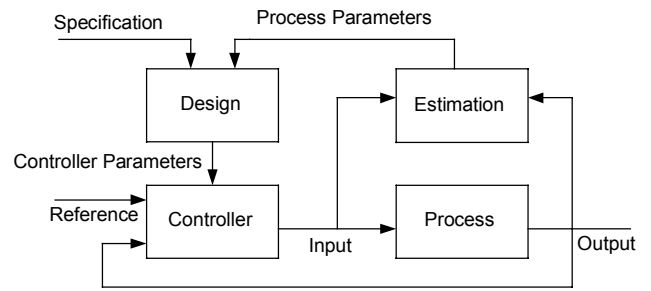


Figure 4 – Block diagram of the adaptive controller.

The controller was implemented in MATLAB.

System Identification

The system identification provides the estimated parameters for the controller design. Pre-identification tests led to the value of the order and dead time of the system. The system's order is two and the dead time one sampling period. The resulting discrete function was obtained using the parametric-type model ARX [6] which is adequate to operate in conjunction with a control function of the pole-placement type. System parameters were estimated using the least squares criterion and a recursive implementation for this method was adopted.

The Control Function

The pole-placement technique was used. With this technique the closed-loop response of the system is totally specified in advance. Closed loop behaviour was determined by choosing adequate values for the damping factor and bandwidth of the system. An observer polynomial was also chosen. The polynomial imposing the closed loop dynamics was chosen so that a bandwidth of 4 rad/s and a damping factor of 1 was obtained. The observer polynomial was chosen to have a bandwidth of 8 rad/s and a damping of 1. The parameters of the control function were obtained by directly solving the resultant Diophantine's equation.

Test Description

Several tests were made under different conditions as reported.

1 - Centralized control

Sensor, controller and actuator functions are centralized in the same node. The block diagram of the system is presented in figure 5.

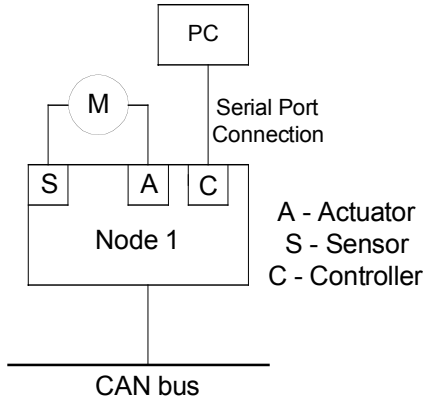


Figure 5 – Block diagram of the system – Test 1.

2 – Half distributed control

The sensor function is implemented by node 1. Controller and actuator functions are implemented using node 2. The block diagram of the system is presented in figure 6.

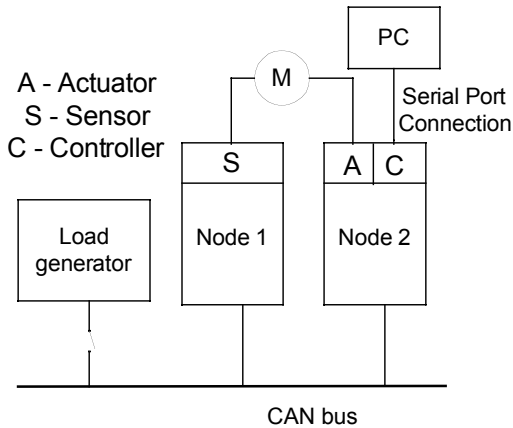


Figure 6 – Block diagram of the system – Test 2.

The sensor node sends a CAN message with the sampled value to the controller/actuator node every sampling period. A block diagram showing the communication scheme is shown in figure 7.

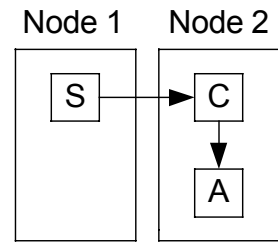


Figure 7 – Communication scheme for test 2.

3 – Fully distributed control

Sensor, controller and actuator are implemented using different nodes, node 1, node 2 and node 3, respectively. The block diagram of the system is shown in figure 8.

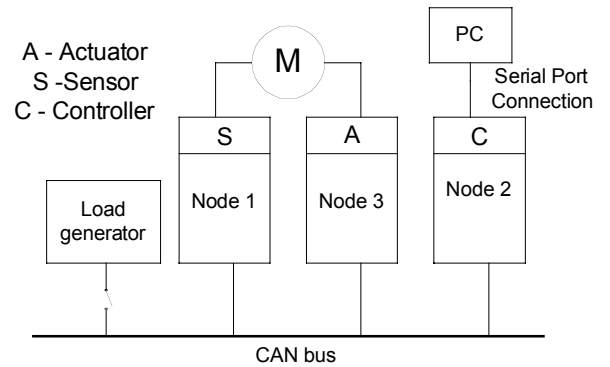


Figure 8 – Block diagram of the system – Test 3.

Two different communication schemes were implemented in this test. The first one uses two different CAN messages: one to send the actuation order directly to the actuator and another sending the sampled value of the process to the controller. Figure 9 presents the communication scheme for Test 3.1.

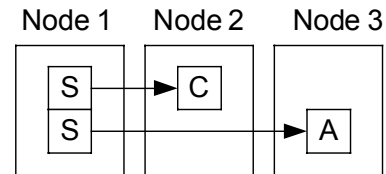


Figure 9 – Communication scheme for test 3.1.

The second communication scheme uses one CAN message from sensor to controller, sending the sampled value and implying the actuation command. The controller gives the actuation order to the actuator sending another CAN

message. Figure 10 presents the communication scheme for Test 3.2.

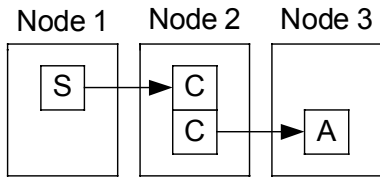


Figure 10 – Communication scheme for Test 3.2.

The tests described were made with and without the load generator node connected to the system. The load generator was intended to generate CAN messages that would introduce jitter in the actuation command of the control scheme. The messages generated used the PSA benchmark [7][8]. Two different patterns for the identifiers of the control messages were introduced: one with the message identifiers lower than the ones of the load messages, and another with message identifiers higher than the ones of the load messages.

Test 1 was intended as a base test and was done without load generation.

The sampling frequency used was 100ms. The identification procedure was stopped after 300 iterations.

Experimental Results

The quadratic mean square error obtained comparing the reference with the output of the

Test			Mean Square Error	
Number	Load	Ids	Best case	Worst case
1	no	----	0.1241	0.1757
2	no	----	0.1301	0.3628
2	yes	low	0.1509	0.8992
2	yes	high	0.1541	0.4049
3.1	no	----	0.1051	0.4349
3.1	yes	low	0.1323	0.3210
3.1	yes	high	0.1458	0.2992
3.2	no	----	0.1435	0.4096
3.2	yes	low	0.1882	0.4270
3.2	yes	high	0.1866	0.3915

Table 1 – Report of the Mean Square Error

system is presented in Table 1 for all the tests. The Ids field refers to the control messages identifiers.

Test 1:

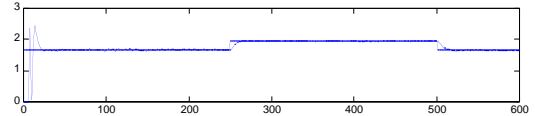


Figure 11 – Experimental results of test 1- best case.

Test 2:

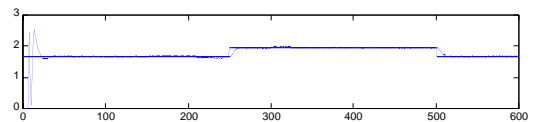


Figure 12 – Experimental results of Test 2 without load generation – best case.

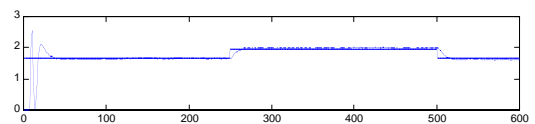


Figure 13 – Experimental results of Test 2 with load generation and low control message Ids – worst case.

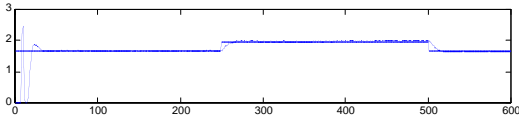


Figure 14 – Experimental results of Test 2 with load generation and high control message Ids – worst case.

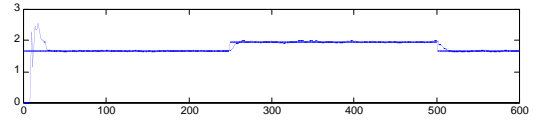


Figure 17 – Experimental results of Test 2 with load generation and high control message Ids – worst case.

Test 3.1:

Test 3.2:

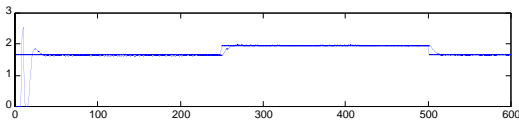


Figure 15 – Experimental results of Test 3.1 without load generation – best case.

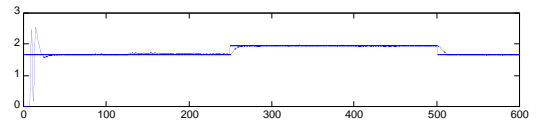


Figure 18 – Experimental results of Test 3.2 without load generation – best case.

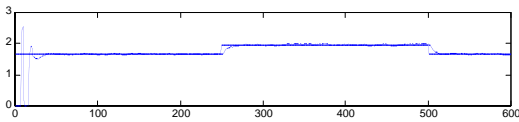


Figure 16 – Experimental results of Test 3.1 with load generation and low control message Ids – worst case.

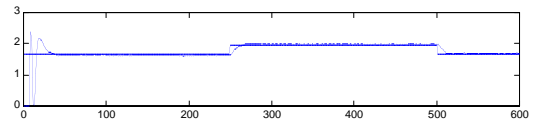


Figure 19 – Experimental results of Test 3.2 with load generation and low control message Ids – worst case.

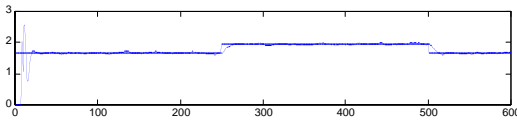


Figure 20 – Experimental results of Test 3.2 with load generation and high control message ids – worst case.

The results show that the introduction of load generation, that will lead to reading and actuation jitter, degrades the quality of the controller system, as does the distribution of the control between several nodes. The degradation of the control systems is not very severe as the control system doesn't lose the ability of following the reference signal. In the present case, the degradation introduced doesn't seem to be related with the control message identifiers as can be worst with lower identifiers (Test 3.1 and Test 3.2) than with higher ones! This can be considered very odd and must be validated with further tests. Anyway, several questions can be posted. In our opinion the first ones are: Could the ratio between the load traffic periods and the sampling period impose this behavior? Can the closed-loop characteristics of the adaptive controller overcome the jitter influence? Is the real system noise hiding the jitter effects? Should we be concerned with the jitter effect when controlling real noisy systems?

Conclusions

A distributed control architecture for adaptive control based in the CAN bus was implemented and experimental data of the degradation of control system was collected. The tests were made under different load conditions and distribution strategies and the results show that the distribution of the system and the introduction of jitter led to a degradation of the control. That degradation is not very severe and the controller never loses the ability to follow the reference signal. As future work we will try to get answers for the posted questions and study other control

strategies under the same conditions to verify if the results depend on the control strategy.

References

- [1] Stothert, et al - «Effect of Timing Jitter on Distributed Computer Control System Performance», Proc. 15th IFAC Workshop DDC'98 – Distrib. Comp. Control Syst., September 1998.
- [2] Alexandre Mota, José A. Fonseca – «Dealing with jitter in Systems Modelling and Identification», Controlo 2000, Guimarães, Portugal, October 2000.
- [3] Alexandre Mota, Pedro Fonseca, José Alberto Fonseca – «Systems Modelling and Identification in CAN based Distributed Control Systems», IFAC, Distributed Computer Control Systems 2000, Sidney, Australia, December 2000.
- [4] P. Fonseca, F. Santos, A. Mota, J.A. Fonseca - «A dynamically reconfigurable CAN system», 5th International CAN Conference, San Jose, California, November 1998.
- [5] Karl J. Åström, Björn Wittenmark – «Adaptive Control», Addison-Wesley, 2nd edition, 1995.
- [6] Lennart Ljung - «System Identification – Theory for the user», Prentice Hall, 1987.
- [7] N. Navet, Y.-Q. Song – «Design of Reliable Real-Time Applications Distributed over CAN (Controller Area Network)», INCOM'98 - IFAC Int. Symp. On Information Control in Manufacturing, Nancy/ Metz, France, 1998.
- [8] Pedro Fonseca – «Modélisation et validation des algorithmes non-déterministes de synchronisation des horloges», PhD Thesis, Institut National Polytechnique de Lorraine / Universidade de Aveiro, 1999.