COMPARING THE DELAY COMPENSATOR APPROACH WITH FRACTIONAL DEAD-TIME MODELLING IN DISTRIBUTED REAL-TIME CONTROL SYSTEMS

Ana Antunes[1], Fernando Morgado Dias[2], Alexandre Mota[3]

[1]ESTSetúbal, Instituto Politécnico de Setúbal
Campus do IPS, Estefanilha, 2910-761 Setúbal, Portugal
[2]DME, Universidade da Madeira
Campus da Penteada, 9000-390 Funchal, Portugal
[3]DETI, Universidade de Aveiro
Campus Universitário de Santiago, 3810-193 Aveiro, Portugal
Email: [1]aantunes@est.ips.pt; [2]morgado@uma.pt; [3]alex@det.ua.pt

Abstract: This paper compares the control performance of a distributed control system with variable sampling to actuation delay when using two different control solutions. The first one is a pole-placement controller with a fuzzy delay compensator and the second one an adaptive pole-placement with fractional dead-time modelling. The delay compensator is added to an existing controller that does not take into account the sampling to actuation delay while the second controller is designed to take that delay into account. The first approach requires the knowledge of the sampling to actuation delay. The results show that both approaches allow the improvement of the control performance. The advantages and disadvantages of each approach are outlined.

Keywords: distributed computer control systems, delay compensation, fuzzy control, adaptive control.

1. INTRODUCTION

Distributed computer control systems are widely used in embedded applications. The distribution of the controller over the network induces variable delays between the sampling instant and the actuation instant. These variable sampling to actuation delays are due not only to the Medium Access Control (MAC) of the network but also to the use of shared resources like the processor and the network by several applications (control or non-control). These delays introduced in the control loop degrade the control performance and may even destabilize the system (Sanfridson, 2000; Colom, 2002; Cervin, 2003; Tipsuwan and Chow, 2003; Antunes, *et al.*, 2004a; Antunes, *et al.*, 2004b).

This paper compares the control performance of a delay compensator based on fuzzy logic, applied to an existing pole-placement controller that ignores the sampling to actuation delay, with a pole-placement controller that takes into account the sampling to actuation delay.

The first control technique requires the knowledge of the sampling to actuation delay that affects the system at each control cycle in order to compute the compensator output while the second control technique does not require that knowledge.

Because the delay compensator approach is based on the knowledge of the effective sampling to actuation delay affecting the system it requires that the controller and the actuator are implemented in the same processing node.

The pole-placement controller with fractional dead-time modelling to model the delay effects does not present this limitation.

The remainder of this paper is organized as follows: section 2 presents the fuzzy delay compensator, section 3 presents the adaptive pole-placement controller using delay modelling, section 4 describes

the test system, section 5 presents the tests and the results obtained and section 6 concludes the paper.

2. THE FUZZY DELAY COMPENSATOR

The delay compensator technique was presented in (Antunes, *et al.*, 2006b). This technique adds a delay compensator to an existing control system that does not take into account the variable sampling to actuation delay. The compensator proposes a correction to the control signal in order to overcome the degradation introduced by the variable sampling to actuation delay.
Figure 1 presents the block diagram of the delay compensator principle.

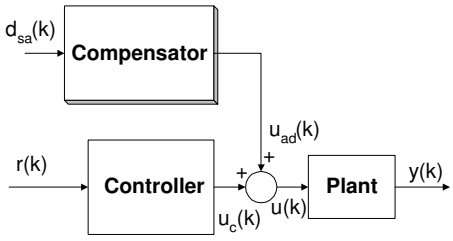

Fig. 1. Block diagram of the delay compensator principle.

The output of the compensator is based on the sampling to actuation delay that affects the system at each control cycle and can have any other input.
This principle can be applied to any distributed control system provided that the sampling to actuation delay is known for each control cycle. The determination of the sampling to actuation delay can be done either by an online measurement or by off-line computations depending on the a priori knowledge of the overall system operation conditions. The online measurement of the sampling to actuation delay provides a more generic and flexible solution that does not require the knowledge of the details of the operation conditions of the global system in which the distributed controller is inserted.
In order for the compensator to operate with the correct value of the delay affecting the network the controller and the actuator must be implemented in the same network node.
The delay compensator principle is generic and can be implemented using different techniques. Since the compensator contribution is added to the control signal output by the existing compensator it can easily be turned on or off and the loop control will be assured by the existing controller. The implementation presented here is based on fuzzy logic and was described in (Antunes, *et al.*, 2006b). Figure 2 presents the block diagram of the fuzzy delay compensator.
The existing pole-placement controller does not take into account the sampling to actuation delay. The controller parameters are constant and computed based on the discrete-time transfer function given by equation (1).

$$G(q^{-1}) = \frac{C\Gamma q^{-1}}{qI - \Phi} \qquad (1)$$

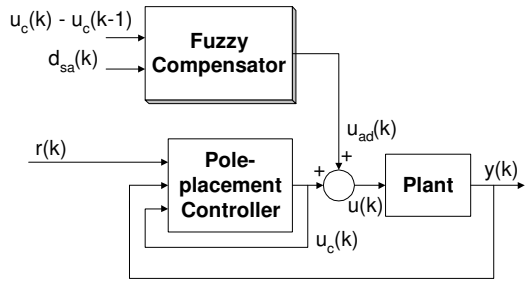

Fig. 2. Block diagram of the fuzzy delay compensator.

Φ and Γ are obtained from the continuous-time system

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t) \qquad (2)$$

$$y(t) = Cx(t) \qquad (3)$$

using the following equations (see Åström and Wittenmark, 1997):

$$\Phi = e^{As} \qquad (4)$$

$$\Gamma = \int_0^h e^{Ah} dsB \qquad (5)$$

where h is the sampling period.
In a control system with sampling to actuation delays the control signal is applied to the plant for less (or more) than the sampling period time. The fuzzy compensator provides a correction to the control signal of the pole-placement controller based on a simple linear approximation of the effect of the sampling to actuation delay.
The fuzzy module uses a function of the Mamdani type (Mamdani and Assilian, 1975) with two inputs (the sampling to actuation delay $d_{sa}(k)$ and the difference between the previous values of the control signal generated by the pole-placement controller $u_c(k)$-$u_c(k-1)$), one output (the compensation value $u_{ad}(k)$) and has six rules.
The rules used state that if the difference between the previous two samples of the control signal is “null” then the contribution of the compensator is “null”. If the delay is “low” then the contribution is “null”. Otherwise if the delay is medium or high, the contribution of the compensator will also be medium or high, with a sign given by the difference between the previous two samples of the control signal: if the control signal is decreasing then the contribution of the compensator is negative and if the control signal is increasing then the contribution is positive.
The membership functions of the inputs and outputs are of the bell type. For details about the implementation and the results obtained consult (Antunes, *et al.*, 2006b).

3. THE ADAPTIVE POLE-PLACEMENT CONTROLLER WITH FRACTIONAL DEAD-TIME MODELLING

The delay compensator performance is compared with the one of a self-tuning regulator of the pole-placement type (like the existing controller) that takes

into account the value of the sampling to actuation delay in each control cycle.
The pole-placement controller parameters are computed at each control cycle and are based on the online estimated parameters of the discrete-time transfer function given by equation (6). This model takes into account the variable sampling to actuation delay by modelling it as a fractional dead-time. The details of this technique are described in (Antunes, *et al.*, 2005) and (Antunes, *et al.*, 2006a).

$$G(q^{-1}) = \frac{C(\Gamma_0 + \Gamma_1 q^{-1})}{qI - \Phi} \tag{6}$$

Γ_0 and Γ_1 are obtained from the continuous-time system given by equations (7) and (8)

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t-\tau) \tag{7}$$

$$y(t) = Cx(t) \tag{8}$$

using the following equations (see Åström and Wittenmark, 1997):

$$\Gamma_0 = \int_0^{h-\tau} e^{As} ds B \tag{9}$$

$$\Gamma_1 = e^{A(t-\tau)} \int_0^{\tau} e^{Ah} ds B \tag{10}$$

where h is the sampling period. Φ is given by equation (4).
For a system of the first order the discrete-time transfer function is given by equation (11)

$$G(q^{-1}) = \frac{b_1 q^{-1} + b_2 q^{-2}}{1 - aq^{-1}} \tag{11}$$

4. THE TEST SYSTEM

This section describes the distributed system architecture, the existing pole-placement controller of the delay compensator approach and the adaptive pole-placement controller that takes into account the variable sampling to actuation delay.

4.1 The distributed system architecture

The distributed control system is composed of two nodes: the sensor node and the controller and actuator node, connected using the CAN bus. The controller and the actuator have to share the same node in order to be possible to measure accurately the value of the sampling to actuation delay that affects the control loop at each control cycle. For the adaptive pole-placement technique the controller and the actuator can be implemented in different nodes because it is not necessary to measure the actual sampling to actuation delay affecting the system.
The block diagram of the distributed system is presented in figure 3.
The sensor samples the plant and sends the sampled value to the controller and actuator node using one message. The controller computes the actuation value based in the control scheme chosen and the actuator acts upon the plant.
The transfer function of the plant is given by equation (12).

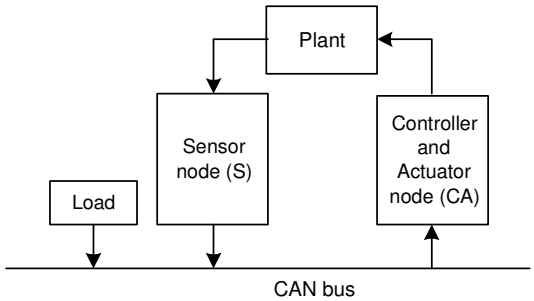

Fig. 3. Block diagram of the distributed system.

$$\frac{Y(s)}{U(s)} = \frac{0.5}{s + 0.5} \tag{12}$$

The sampling period is equal to 280ms and was chosen according to the rule of thumb proposed by (Åström and Wittenmark, 1997). The sampling period is the same for both pole-placement controllers.
The fuzzy compensator was implemented using the evalfis function of the MATLAB fuzzy toolbox.

4.2 Existing controller of the delay compensator approach

The existing controller is of the pole-placement type. The pole-placement technique allows the complete specification of the closed-loop response of the system by the appropriate choice of the poles of the closed-loop transfer function.
In this case the closed-loop pole is placed at α_m=2Hz and an observer was placed at α_0=4Hz.
The parameters of the discrete-transfer function given by equation (1) were computed off-line using equations (4) and (5), Φ= 0.87 and Γ=0.26.
The parameters of the control function are constant and were obtained by directly solving the Diophantine equation for the system. The control function is given by equation (13).

$$u_c(k) = t_0(r(k) - a_o r(k-1)) - s_0 y(k) - s_1 y(k-1) + u_c(k-1) \tag{13}$$

where t_0=3.2832, a_o= 0.3263, s_0=7.4419 and s_1= -5.2299.

4.3 Pole-placement controller with fractional dead-time modelling

The closed-loop pole and the observer were placed at α_m=2Hz and α_0=4Hz, respectively, like the ones of the existing controller of the delay compensator approach in order to be possible to compare the results obtained.
The parameters a, b_1 and b_2 of the discrete-time transfer function given by equation (11) were estimated using a recursive implementation of the least square criterion with forgetting factor based on

the parametric-model ARX (Auto-Regressive with an eXogenous signal) (Ljung, 1987) to model the discrete-time function. This model is appropriate to operate with a control function of the pole-placement type. The forgetting factor is equal to 0.99.
The parameters of the control function are computed online and were obtained by directly solving the Diophantine equation for the system. The control function is given by equation (14).

$$u(k) = t_{0d}(r_{ef}(k) - a_{obs} r_{ef}(k-1)) - s_{0d} y(k) - s_{1d} y(k-1) \ldots$$
$$-(r_{1d} - 1)u(k-1) + r_{1d} u(k-2) \qquad (14)$$

5. TESTS AND RESULTS

The system under test corresponds to the architecture presented in figure 3.
The system was simulated using TrueTime, a MATLAB/Simulink based simulator for real-time distributed control systems (Cervin, *et al.*, 2002; Henriksson and Cervin, 2004).
This work reports three different tests. Test 1 is the reference test, where the sampling to actuation delay is constant and equal to 4ms. It corresponds to the minimum value of the MAC and processing delays. Test 1 was performed using the pole-placement controller that does not consider the sampling to actuation delay (PP) alone, with the fuzzy compensator added to the PP (FDC) and with the pole-placement controller that takes into account the sampling to actuation delay (PPD). The control signals obtained for test 1 using the PP controller are presented in figure 4. The ones obtained with the FDC and the PPD controllers are similar.

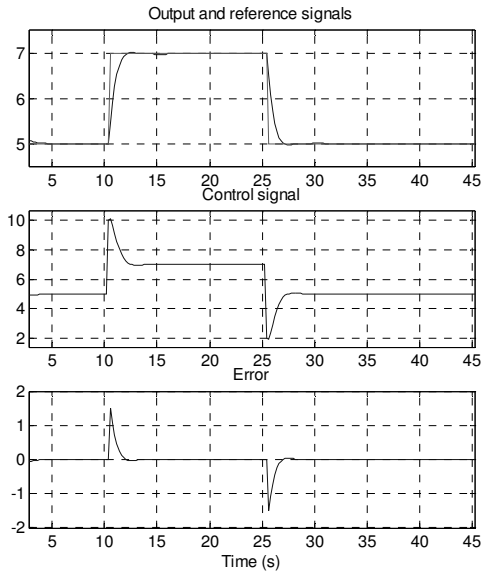

Fig. 4. Control signals for test 1, PP controller.

In tests 2 and 3 additional delay was introduced to simulate a loaded network. The sampling to actuation delay introduced follows a random distribution over the interval [0,h] for test 2 and a sequence based in the gamma distribution that concentrates the values in the interval [h/2, h], the upper half of the sampling period, for test 3.

The sampling to actuation delay obtained for tests 2 and 3 is depicted in figure 5 and 6.

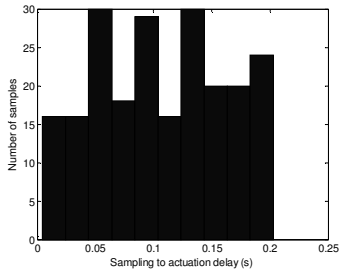

Fig. 5. Sampling to actuation delay for test 2.

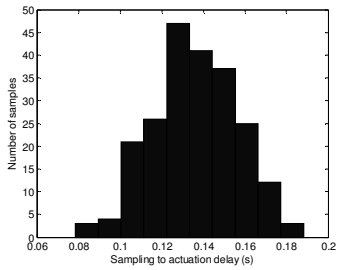

Fig. 6. Sampling to actuation delay for test 3.

The control performance was assessed by the computation of the Integral of the Squared Error (ISE) between t= 8s and 45s. The results obtained for ISE are presented in Table 1.

Table 1 ISE report.

Test	PP	FDC	PPD
1	3.3	3.3	3.4
2	3.7	3.5	3.6
3	3.9	3.6	3.8

The percentage of improvement obtained by report to the reference test for ISE is presented in Table 2.
The improvement is calculated as the amount of error induced by the sampling to actuation delay that the FDC or PPD controllers were able to reduce. The formula used for the computation of the improvement is presented in (15).

$$Iprv(\%) = (1 - \frac{ISE_D - ISE_{D\,\mathrm{Ref}}}{ISE_{PP} - ISE_{PP\,\mathrm{Ref}}}) * 100 \qquad (15)$$

Table 2 Improvement for ISE.

Test	FDC	PPD
2	50%	50%
3	50%	33%

The control signals for test 2 and 3 using the PP, FDC and PPD controllers are presented in figures 7 to 12.
The results show that variable sampling to actuation delay concentrated in the interval [h/2, h] results in a bigger degradation of the control performance when compared with a variable sampling to actuation delay randomly distributed over the interval [0,h].
As expected when analysing the ISE report the PP controller presents the worst results because it does not take into account the sampling to actuation delay. Both the FDC and the PPD controllers present better control performance than the PP controller.

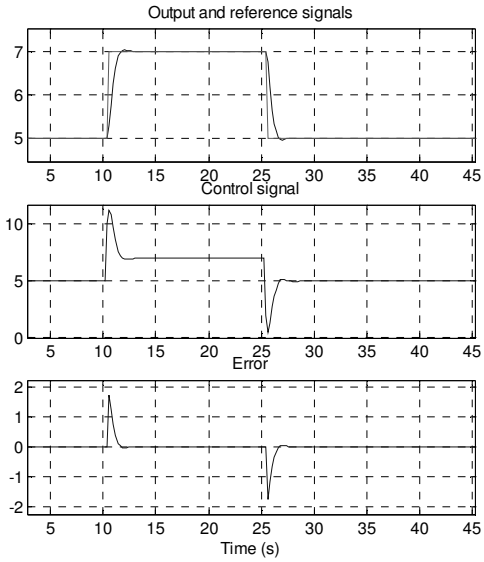

Fig. 7. Control signals for test 2, PP controller.

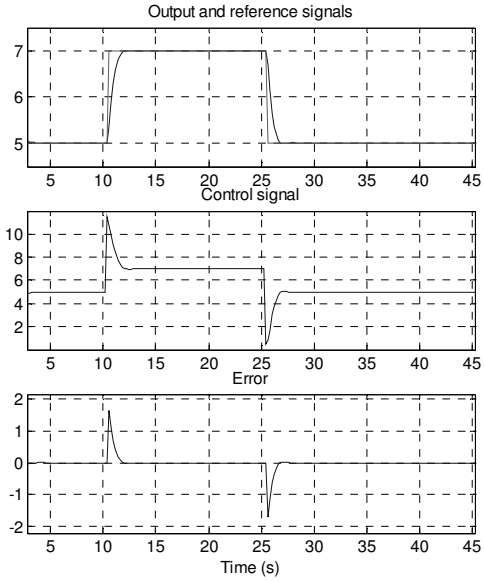

Fig. 8. Control signals for test 2, FDC controller.

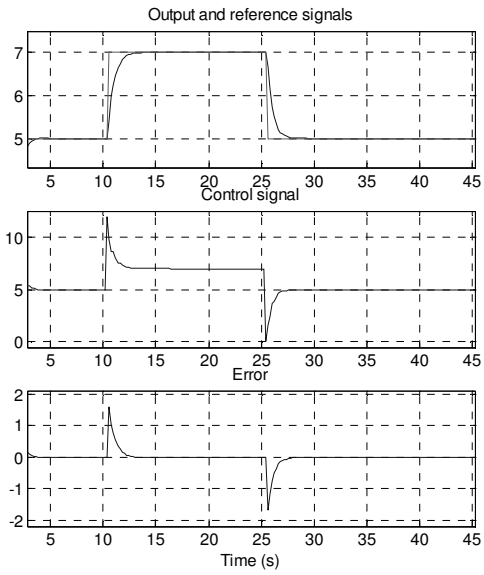

Fig. 9. Control signals for test 2, PPD controller.

The FDC controller allows an improvement of 50% for both test 2 and test 3, while the PPD controller allows an improvement of 50% for test 2 and 33% for test 3.

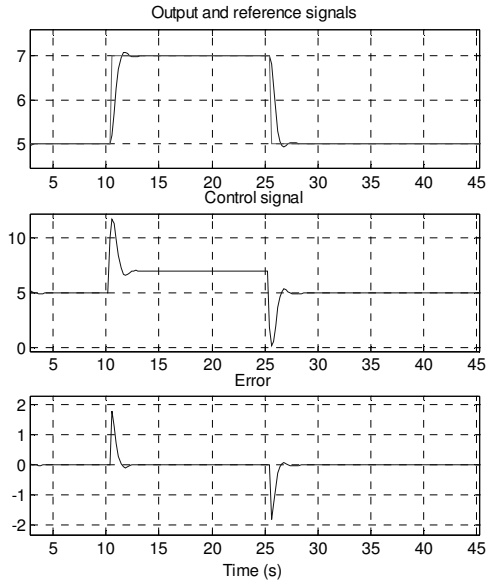

Fig. 10. Control signals for test 3, PP controller.

The fuzzy delay compensator follows the control signal of the PP controller proposing a correction value according to the amount of the sampling to actuation delay but requires the knowledge of the delay that affects the system. The online measurement of the sampling to actuation delay in a distributed control system is not straightforward in most of the cases.

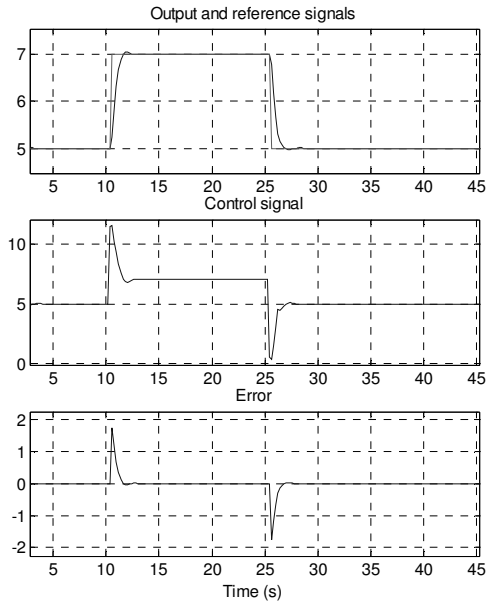

Fig. 11. Control signals for test 3, FDC controller.

The PPD controller is also able to improve the degradation introduced by the sampling to actuation delay and does not require the measurement of the sampling to actuation delay.

It is also interesting to have an idea of the execution times of the control algorithms in study. The FDC

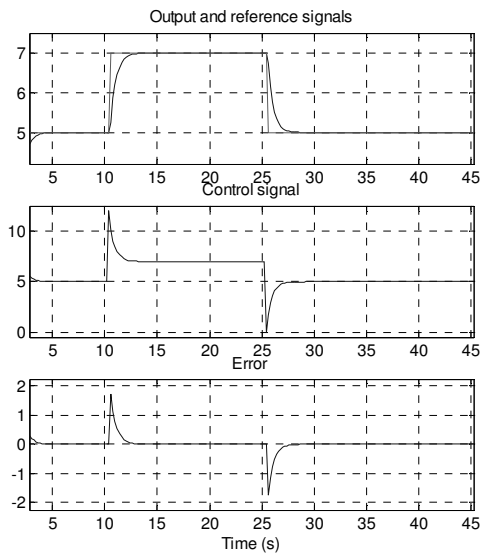

Fig. 12. Control signals for test 3, PPD controller.

controller executes in 7ms and the PPD controller in 1ms. The execution times were obtained using the tic and toc MATLAB functions. The PPD controller is 86% faster than the FDC approach.

6. CONCLUSION

This paper presents the results of the comparison between two different control schemes to use in distributed computer control systems.

It compares a fuzzy delay compensator scheme that is added to an existing pole-placement controller (that does not take into account the variable sampling to actuation delay) in order to improve the control performance, with an adaptive pole-placement controller that takes that delay into account by modelling it as fractional dead-time.

The results show that taking into account the variable sampling to actuation delay when designing the pole-placement controller allows better results than ignoring it. This technique does not require the measurement of the sampling to actuation delay and consequently does not introduce restrictions on the implementation of the distributed control system. It also results in a faster algorithm.

The fuzzy delay compensator is also able to improve the control performance and can be easily added to an existing controller that does not take into account the variable sampling to actuation delay. It requires the measurement of the sampling to actuation delay and that conditions the implementation of the distributed system. The algorithm used to implement the FDC technique is slower than the one used for the PPD controller.

REFERENCES

Antunes, A. F.M. Dias and A.M. Mota. (2004a). Influence of the sampling period in the performance of a real-time adaptive distributed system under jitter conditions, *WSEAS Transactions on Communications*, **3**, pp. 248-253.

Antunes, A. and A.M. Mota, (2004b). Control Performance of a Real-time Adaptive Distributed Control System under Jitter Conditions, Proc. Conf. Control 2004, Bath, UK.

Antunes, A., F.M. Dias and A.M. Mota. (2005) Fractional dead-time modelling for variable sampling to actuation delay compensation in distributed real-time control systems, *International Transactions on Computer Science and Engineering*, GESTS, **24**, nº 1, pp. 9-18.

Antunes, A., F.M. Dias and A.M. Mota. (2006a) Improving the control performance of a real-time distributed control system under variable sampling to actuation delay, Proc. 25th IASTED International Conference on Modelling, Identification and Control, Lanzarote, Lanzarote, Spain.

Antunes, A., F.M. Dias, J. Vieira and A Mota. (2006b) Delay Compensator: an approach to reduce the variable sampling to actuation delay effect in distributed real-time control systems, Proc. 11th IEEE International Conference on Emerging Technologies and Factory Automation, Prague, Czech Republic.

Åström, K.J. and B. Wittenmark. (1997). *Computer Controlled Systems*, Prentice Hall.

Cervin, A., D. Henriksson, B. Lincoln, and K.-E. Årzén. (2002) Jitterbug and TrueTime: analysis tools for real-time control systems, Proc. 2nd Workshop on Real-Time Tools, Copenhagen, Denmark.

Cervin, A. (2003). Integrated control and real-time scheduling, Ph.D. dissertation, Lund Institute of Technology.

Colom, P.M. (2002). Analysis and design of real-time control systems with varying control timing constraints, Ph.D. dissertation, Universitat Politecnica de Catalunya.

Henriksson, D., and A. Cervin. (2004) TrueTime 1.2-Reference Manual", Dept. of Automatic Control, Lund Institute of Technology, Sweden.

Ljung, L. (1987) *System identification- Theory for the user*, Prentice Hall.

Mamdami, E.H. and S. Assilian. (1975) An experimental in linguistic synthesis with a fuzzy logic controller, *International Journal of Man-Machine Studies*, **7**, pp. 1-13.

Sanfridson, M. (2000). Timing problems in distributed real-time computer control systems, Tech. Rep. ISSN 1400-1179, Royal Institute of Technology, Sweden.

Tipsuwan, Y. and M.-Y. Chow. (2003). Control methodologies in networked control systems, *Control Engineering Practice*, **11**, pp. 1099-1111.