

# Optimal On-line Sampling Period Assignment for Real-Time Control Tasks Based on Plant State Information

Dan Henriksson and Anton Cervin

**Abstract**—The paper presents a feedback scheduling strategy for multiple control tasks that uses feedback from the plant states to distribute the computing resources optimally among the tasks. Linear-quadratic controllers are analyzed, and expressions relating the expected cost to the sampling period and the plant state are derived and used for on-line sample-rate adjustments. In the case of minimum-variance control of multiple integrator processes, an exact expression for the optimal sampling periods is obtained. For the general case, an on-line optimization procedure is developed. The approach is exemplified on a set of controllers for first-order systems. The issues of computational delay and the choice of the feedback scheduler period are also discussed.

## I. INTRODUCTION

In embedded systems, the computational resources are generally limited and must be used as efficiently as possible. At the same time, product developers want to add more and more functionality to the systems. Consequently, several concurrent activities are typically competing for the limited resources. It is therefore desirable to have methods that optimize the performance of control loops implemented in systems with scarce computing resources.

Embedded control systems have traditionally been relatively static and have been developed assuming fixed sampling rates and fixed task periods. The benefit of this design methodology is that it has enabled the application of standard sampled-data control theory (e.g., [1]) and standard real-time scheduling theory (e.g., [2]). However, in recent years dynamic approaches to real-time control have been receiving increased attention. Relaxing the classical assumptions on static task sets with fixed periods and fixed computation times can give potentially higher resource utilization and better control performance. On the other hand, such approaches tend to make both the control and the real-time designs more complicated.

The key to successful design and implementation of flexible embedded real-time control systems is *co-design*. By considering the design of the controller and its real-time implementation as integral parts, instead of in isolation, better and more resource-efficient systems can be obtained. One co-design method that aims at increasing flexibility and adapting to variations in task execution patterns is *feedback scheduling* [3]. The objective of a feedback scheduler is to distribute computing resources in a way that optimizes the

This work has been supported by the research projects SSF/FLEXCON, EU/ARTIST2, and EU/RUNES.

D. Henriksson and A. Cervin are with the Department of Automatic Control LTH, Lund University, Box 118, SE 221 00 Lund, Sweden. Email: {dan, anton}@control.lth.se

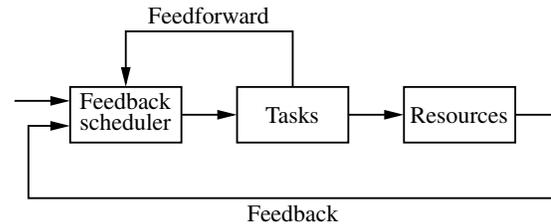


Fig. 1. A general feedback scheduling structure for regulation of limited computer resources.

overall control performance, subject to constraints on the total system workload.

A general feedback scheduling structure is shown in Fig. 1. The scheduler uses feedback from the resource consumption (typically execution times), but it may also use feedback from the actual control performance of the different loops. The reactive feedback may also be combined with pro-active feedforward actions, in response to, e.g., mode changes affecting the execution times [4]. The manipulated variables are usually task periods or priorities, but for some controllers it may also be possible to change execution times, e.g., [5].

In this paper, we consider the problem of optimal on-line assignment of sampling periods for a set of linear-quadratic (LQ) controllers. The presentation will focus on scheduler decisions based on feedback from the actual control performance, i.e., the current plant states, rather than on resource consumption. By developing cost functions that depend on the plant states at the start of the feedback scheduler period, resources can be distributed in a way that reflects the current control performance. Hence, disturbances affecting some control loops will induce the corresponding tasks to run at a faster rate than the loops that are in equilibrium.

The task period assignment is based on finite-horizon cost functions related to the sampling period and to the current plant state. The expressions for the cost functions are developed analytically for the case of integrator processes. For general systems, an on-line optimization procedure is developed. The feedback scheduling approach is evaluated in a number of examples, where co-simulation of the plants, the control tasks, and the feedback scheduler is used.

### A. Related Work

Seto *et al.* [6] first introduced the idea of control performance optimization under system utilization constraints. A performance index expressed as a function of the sampling rate was used as a basis for the optimization.

Eker *et al.* [7] developed a feedback scheduler for LQ

controllers, which used sampling period adjustments to regulate the CPU load. However, the scheme did not involve feedback from the plant states and hence did not incorporate the actual performance of the individual control loops.

Martí *et al.* [8] presented the idea of dynamic resource allocation based on state feedback from the controlled plants. This work, however, lacks a theoretical foundation in the definition of the state-dependent cost functions.

Static, optimal switching sequences for a number of linear quadratic controllers were considered in [9]. Only one controller may be accessed each time-slot, e.g., motivated by networked control loops with limited bandwidth. This can be seen as a variable sample rate problem where the sampling rates are multiples of the time-slot.

Event-based control of first-order systems was considered in [10]. Expressions relating the LQ cost to the sampling interval for first-order systems were given, assuming an infinite optimization horizon and zero control signal penalty.

Stability properties for systems with varying sampling rate were investigated in [11]. Here it was shown that certain switching sequences between two LQ-controllers designed with different sampling intervals may cause instability, although the individual controllers were stable. A stabilizing optimal control law was also presented and analyzed.

Also, many feedback-based scheduling algorithms, see, e.g., [12], have been proposed in the real-time scheduling community. However, our concept of feedback scheduling is fundamentally different from their objectives, which mainly aim at obtaining a desired deadline miss ratio for the tasks with no connection to system performance.

## B. Outline of Paper

The rest of this paper is organized as follows. Section II presents the problem formulation. Section III treats the case of an integrator process and solves the optimization problem analytically for the minimum variance controller. The problem is extended to general first-order systems in Section IV. Here the cost functions are linearized to facilitate on-line optimization. Section V discusses implementation issues and the paper is concluded by Section VI.

## II. PROBLEM FORMULATION

We consider a situation where a number of controller tasks share the same CPU, and therefore need to be scheduled. Since the tasks implement controllers, the objective of the scheduler is to distribute computing resources in a way that maximizes the overall control performance.

The idea is to apply feedback at two levels in the implementation of the real-time control system, see Fig. 2. Apart from the standard feedback used by the controller tasks, the second level represents feedback within the real-time system to dynamically allocate computing resources between competing tasks based on information about the current states of the controlled plants.

We consider a set of periodic controller tasks designed using LQ control theory. We further allow the sampling periods of the controllers to be changed within a dynamic

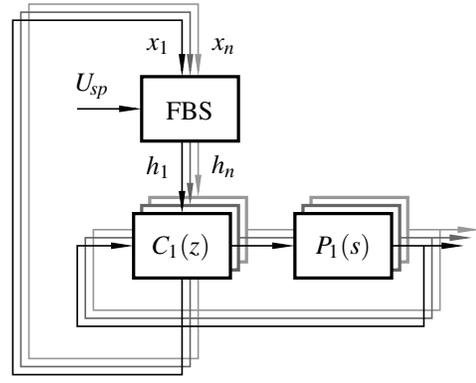


Fig. 2. Dynamic resource allocation in multiple control loops using a feedback scheduler (FBS). The scheduler assigns sampling periods  $h_1 \dots h_n$ , based on the plant states  $x_1 \dots x_n$ , to meet the utilization setpoint  $U_{sp}$ .

range as decided by the feedback scheduler. The scheduler runs as a periodic activity with period  $T_{fbs}$ .

### A. The LQ Control Problem

Each plant is assumed to be described by a linear stochastic differential equation

$$dx = Axdt + Budt + dv_c, \quad (1)$$

where  $x$  is the plant state,  $u$  is the control signal, and  $v_c$  is a Wiener process with zero mean and incremental variance  $R_{1c}$ . The initial state of the plant is assumed to be known and equal to  $x(0) = x_0$ . Sampling the plant with the hold interval  $h$  gives a discrete-time system of the form

$$x(k+1) = \Phi(h)x(k) + \Gamma(h)u(k) + v(k) \quad (2)$$

where  $v$  is a discrete-time white noise process with variance  $R_1(h)$  (see [1] for details).

The objective of the control is to minimize a continuous-time cost function,

$$J = \mathbb{E} \left\{ \int_0^{T_{fbs}} \left( x^T(t) Q_{1c} x(t) + 2x^T(t) Q_{12c} u(t) + u^T(t) Q_{2c} u(t) \right) dt + x^T(T_{fbs}) Q_0 x(T_{fbs}) \right\}, \quad (3)$$

where  $Q_{1c}$ ,  $Q_{12c}$ , and  $Q_{2c}$  are design parameters, while  $Q_0$  is chosen as solution to the algebraic Riccati equation (7) below. This “bootstrap” choice of  $Q_0$  ensures that the resulting control law will be time-invariant,

$$u(k) = -L(h)x(k), \quad (4)$$

even though we consider a finite-horizon LQ problem.

The sampled version of the cost function (3) has the form

$$J = \mathbb{E} \left\{ \sum_{k=0}^{N-1} \left( x^T(k) Q_1(h) x(k) + 2x^T(k) Q_{12}(h) u(k) + u^T(k) Q_2(h) u(k) + J_v(h) \right) + x^T(N) Q_0 x(N) \right\}, \quad (5)$$

where  $N = T_{fbs}/h$  is assumed to be an integer, and  $J_v$  is an additional cost term due to the inter-sample noise [13].

Solving the LQ problem gives the optimal cost

$$J = x_0^T \bar{S}(h) x_0 + \sum_{k=0}^{N-1} \left( \text{tr} \bar{S}(h) R_1(h) + J_v(h) \right), \quad (6)$$

where  $\bar{S}$  is the solution to the algebraic Riccati equation

$$\bar{S} = \Phi^T \bar{S} \Phi + Q_1 - (\Phi^T \bar{S} \Gamma + Q_{12}) (\Gamma^T \bar{S} \Gamma + Q_2)^{-1} (\Gamma^T \bar{S} \Phi + Q_{12}^T). \quad (7)$$

Note that  $\bar{S}$ ,  $R_1$ , and  $J_v$  all depend on the sampling interval  $h$ . Further, we note that the cost (6) can be written as a function

$$J(x_0, h, T_{fbs}) = x_0^T \bar{S}(h) x_0 + T_{fbs} \bar{J}(h) \quad (8)$$

where

$$\bar{J}(h) = \frac{\text{tr} \bar{S}(h) R_1(h) + J_v(h)}{h} \quad (9)$$

can be interpreted as the stationary cost per time unit of the controller.

*Remark:* It is also possible to evaluate the cost function (5) for an arbitrary (i.e., non-optimal) state-feedback control law  $u(k) = -Lx(k)$ . In this case, assuming that the closed-loop system is stable, the expression for  $\bar{S}$  in (6), (8), and (9) is replaced by the solution  $\bar{S}$  to the Lyapunov equation

$$(\Phi - \Gamma L)^T \bar{S} (\Phi - \Gamma L) - \bar{S} + Q_1 - Q_{12} L - L^T Q_{12}^T + L^T Q_2 L = 0 \quad (10)$$

Again, this assumes that  $Q_0$  is chosen as  $\bar{S}$ .

### B. The Period Assignment Problem

We now assume that  $n$  control tasks share the same processor. Each task  $i = 1 \dots n$  is described by a constant execution time  $C_i$ , an adjustable period  $h_i$ , and a cost function  $J_i(x_0, h, T_{fbs})$ . When invoked at time  $t$ , the feedback scheduler is informed about the plant state vectors  $x_1(t) \dots x_n(t)$ . It should then assign new sampling intervals  $h_1 \dots h_n$  such that the total expected cost over the next  $T_{fbs}$  units of time is minimized. This is formulated as the following optimization problem:

$$\begin{aligned} \min_{h_1 \dots h_n} \quad & \sum_{i=1}^n J_i(x_i(t), h_i, T_{fbs}) \\ \text{subj. to} \quad & \sum_{i=1}^n \frac{C_i}{h_i} \leq U_{sp} \end{aligned} \quad (11)$$

Here, a suitable choice of utilization setpoint,  $U_{sp}$ , depends on the fraction of the processor available for control and on the scheduling algorithm (see, e.g., [2]).

In [6] it was noted that the optimization problem is convex if the cost functions  $J_i(1/h)$  are convex. In [7], [4] it was shown that the optimization problem has an explicit solution if all cost functions can be described as quadratic functions of  $h$ ,

$$J_i(h) = \alpha_i + \beta_i h^2, \quad (12)$$

or as linear functions of  $h$ ,

$$J_i(h) = \alpha_i + \gamma_i h. \quad (13)$$

In the latter case, the optimal periods are given by

$$h_i = \sqrt{\frac{C_i}{\gamma_i} \frac{\sum_{j=1}^n \sqrt{C_j \gamma_j}}{U_{sp}}} \quad (14)$$

In general, however, the cost functions are not described exactly by (12) or (13). The optimization problem can then be solved iteratively as follows. First, the cost functions are linearized around the current sampling periods to approximate (13). This implies setting

$$\gamma_i = x_i^T(t) \frac{\partial \bar{S}(h_i)}{\partial h} x_i(t) + T_{fbs} \frac{\partial \bar{J}(h_i)}{\partial h} \quad (15)$$

Second, new periods are computed by (14). These steps may then be repeated to refine the solution. To use the optimization routine in practice, the gradients  $\frac{\partial \bar{S}(h)}{\partial h}$  and  $\frac{\partial \bar{J}(h)}{\partial h}$  must be computed off-line and stored in look-up tables.

### III. THE INTEGRATOR CASE

We first consider an integrator process, since this allows us to come up with a closed-form expression for the cost function (6). This is then used to find a solution to the optimization problem (11). The plant is described by

$$dx = u dt + dv_c, \quad (16)$$

where  $x$  is the scalar state,  $u$  is the control signal, and  $v_c$  is a Wiener process with incremental variance  $R_{1c} = 1$ . The LQ controller is designed to minimize the continuous-time cost function (3) with  $Q_{1c} = 1$ ,  $Q_{12c} = 0$ , and  $Q_{2c} = \rho$ . Equation (16) is sampled with the period  $h$  to yield the discrete-time system

$$x(k+1) = x(k) + hu(k) + v(k) \quad (17)$$

Computing the sampled version of the cost function (3) gives

$$Q_1(h) = h, \quad Q_{12}(h) = \frac{h^2}{2}, \quad Q_2(h) = \frac{h^3}{3} + \rho h. \quad (18)$$

Further,

$$J_v = \int_0^h Q_{1c} \int_0^t R_{1c} d\tau dt = h^2/2 \quad (19)$$

and

$$R_1 = E v^2(k) = \int_0^h R_{1c} d\tau = h \quad (20)$$

The stationary Riccati equation for the LQ problem is given by

$$\bar{S} = \bar{S} + Q_1 - \frac{(h\bar{S} + Q_{12})^2}{h^2 \bar{S} + Q_2} \quad (21)$$

Finally, solving for  $\bar{S}$  gives the costs

$$\bar{S} = \sqrt{\frac{h^2}{12} + \rho}, \quad \bar{J} = \bar{S} + \frac{h}{2} \quad (22)$$

As seen, the cost function is generally a non-linear functions of the sampling interval. However, two interesting special cases can be studied.

First, we consider the case when  $\rho \gg h$ . In this case  $\bar{S} \approx \sqrt{\rho}$ , which inserted in (6) gives

$$J(x_0, h, T_{fbs}) \approx \sqrt{\rho} (x_0^2 + T_{fbs}) + \frac{T_{fbs}}{2} h, \quad (23)$$

which is on the form (13). In this case, however, the initial state,  $x_0$ , only appears in the  $\alpha_i$  parameter and does not enter the solution (14).

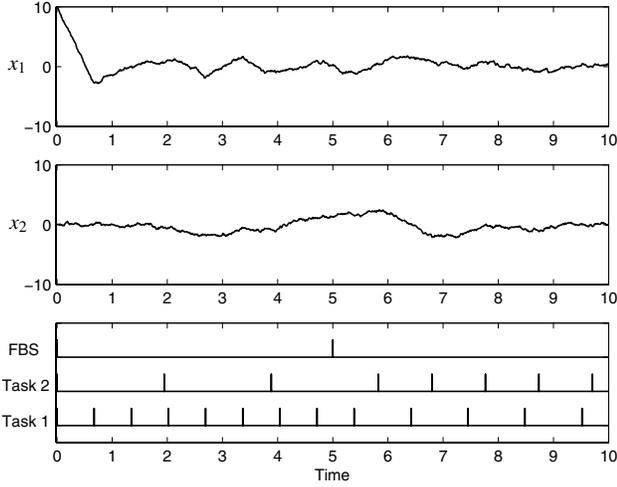


Fig. 3. Simulation of two-integrator system. The initial state of plant 1 causes control task 1 to be executed more often in the first feedback scheduling interval.

The second interesting case is when  $\rho = 0$  (i.e., minimum-variance control). Equation (22) now gives  $S(h) = \sqrt{3}h/6$  and substituting in (6) the cost function becomes

$$J(x_0, h, T_{fbs}) = \left( x_0^2 \frac{\sqrt{3}}{6} + T_{fbs} \frac{\sqrt{3} + 3}{6} \right) h \quad (24)$$

In this case, the cost for each individual controller is an exact linear function of the sampling period,  $h$ , and with a slope that depends on the current state,  $x_0$ .

#### Simulation Example

As an example, we consider concurrent minimum-variance control of two integrator processes (16). Combining (14) and (24), we see that the optimal ratio between the sampling periods becomes

$$\frac{h_2}{h_1} = \sqrt{\frac{C_2(x_1(t)^2 + T_{fbs}(1 + \sqrt{3}))}{C_1(x_2(t)^2 + T_{fbs}(1 + \sqrt{3}))}} \quad (25)$$

A simulation model of the system, including the feedback scheduler, was created using MATLAB/Simulink and the TrueTime toolbox [14]. In the simulation, the plants are disturbed by bandwidth-limited white noise processes with intensity 1. The results of a simulation assuming  $x_1(0) = 10$ ,  $x_2(0) = 0$ ,  $C = 0.5$ ,  $U_{sp} = 1$ , and  $T_{fbs} = 5$  are shown in Fig. 3. It is seen that the initial state  $x_1(0) \neq 0$  causes the first controller to be executed more frequently during the first feedback scheduler period,  $h_1 = 0.67$  and  $h_2 = 1.94$ . At the second feedback scheduler invocation, the plant states are nearly equal, and the assigned periods are hence also similar:  $h_1 = 1.03$  and  $h_2 = 0.97$ .

Note that, in this simulation, the computational delay in the controllers has been neglected. This aspect will be studied in Section V.

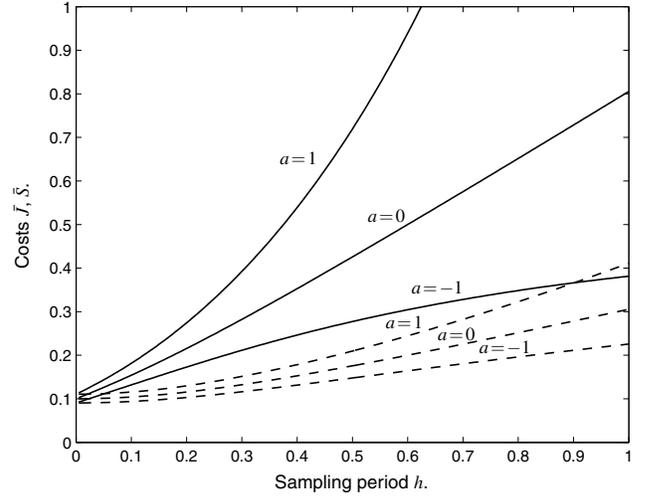


Fig. 4. Costs for different first-order systems with  $\rho = 0.01$ : Stationary cost per time unit,  $\bar{J}$  (full); cost due to the initial state,  $\bar{S}$  (dashed).

#### IV. FIRST-ORDER SYSTEMS

General first-order systems are now considered. Let the plant be described by

$$dx = ax dt + u dt + dv_c \quad (26)$$

where  $x$  is the scalar state,  $u$  is the control signal, and  $v_c$  is a Wiener process with incremental variance  $R_{1c} = 1$ . The sampled version of (26) is given by

$$x(k+1) = e^{ah}x(k) + \frac{1}{a}(e^{ah} - 1)u(k) + v(k) \quad (27)$$

Again, the LQ controller is designed to minimize the continuous-time cost function (3) with  $Q_{1c} = 1$ ,  $Q_{12c} = 0$ , and  $Q_{2c} = \rho$ . The sampled version of the cost function gives

$$\begin{aligned} Q_1(h) &= \frac{e^{2ah} - 1}{ah} \\ Q_{12}(h) &= \frac{1}{2a^2}(e^{2ah} - 2e^{ah} + 1) \\ Q_2(h) &= \frac{1}{2a^3}(2ah + e^{2ah} - 4e^{ah} + 3) + \rho h \end{aligned} \quad (28)$$

Further,

$$R_1(h) = \int_0^h e^{2a\tau} R_{1c} d\tau = \frac{1}{2a}(e^{2ah} - 1) \quad (29)$$

and

$$J_v(h) = \int_0^h Q_{1c} \int_0^t e^{2a\tau} R_{1c} d\tau dt = \frac{1}{4a^2}(e^{2ah} - 2ah - 1) \quad (30)$$

In this case, the solution to the Riccati equation (21) becomes quite complicated and is not given here. Fig. 4 plots the functions  $\bar{S}(h)$  and  $\bar{J}(h)$  for  $\rho = 0.01$  and  $a = -1, 0, 1$ . It is seen that the costs are nonlinear in  $h$ , and increase more rapidly for the unstable process,  $a = 1$ . Locally, however, the costs can be approximated quite well by linear functions.

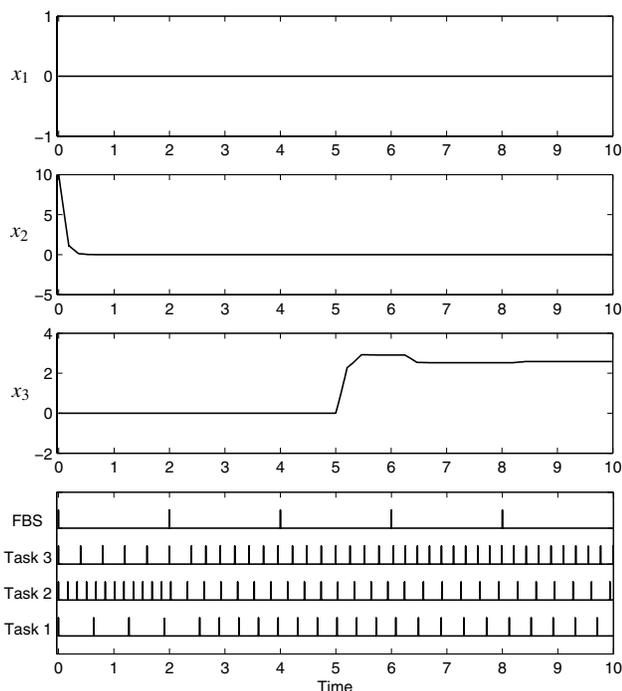


Fig. 5. Optimal feedback scheduling of three first-order systems.

### Simulation Example

As a second example, we consider concurrent control of three first-order plants (26) with  $a = -1$ ,  $a = 0$ , and  $a = 1$ . In the control design, the plants are all assumed to be disturbed by unit-intensity white noise processes. The cost function derivatives  $\frac{\partial \bar{S}(h)}{\partial h}$  and  $\frac{\partial \bar{J}(h)}{\partial h}$  for each controller are computed off-line and stored in look-up tables (six in total).

In the actual simulation, plants 1 and 2 are undisturbed, while a load disturbance of magnitude 10 enters plant 3 at time  $t = 5$ . The results assuming  $x_1(0) = 0$ ,  $x_2(0) = 10$ ,  $x_3(0) = 0$ ,  $C = 0.1$ ,  $U_{sp} = 1$ , and  $T_{fbs} = 2$  are shown in Fig. 5. It is seen that both the initial condition of plant 2 and the load disturbance entering plant 3 causes the sampling periods to be changed by the feedback scheduler.

The results in Fig. 5 were obtained using the exact on-line solutions to the optimization problem (11). This is not realistic for embedded systems, where the resources are already very limited. In a second simulation, the approximate method using linearization of the cost functions was used instead. The current slopes of the cost functions were computed using (15) and then used in (14) to obtain the new periods. The resulting assigned periods are compared to the optimal periods in Fig. 6. It is seen that the linearization method gives quite good results in this example.

## V. REAL-TIME IMPLEMENTATION ISSUES

In the previous sections we have shown how to dynamically assign sampling periods to a set of controller tasks based on run-time information about the performance of each control loop and the current workload. However, from a real-time implementation standpoint some more issues must be taken care of in order to use the strategy in a real setup.

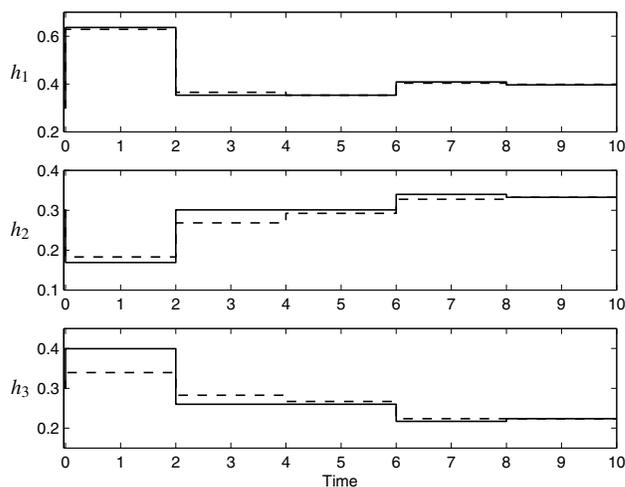


Fig. 6. Assigned sampling periods: optimal solution (full) and approximate solution (dashed).

### A. Choice of $T_{fbs}$

One important design parameter with consequences for the real-time implementation is the feedback scheduler period,  $T_{fbs}$ . Obviously, the shorter the feedback scheduler period, the faster the response to external disturbances affecting the controlled plants. Also, the influence of the initial state on the optimization problem will increase with smaller  $T_{fbs}$ .

On the other hand, the drawback of choosing too short  $T_{fbs}$  is that the execution of the feedback scheduler induces overhead and consumes CPU time from the controller tasks. Consequently, the choice of  $T_{fbs}$  is a trade-off between computational overhead of the feedback scheduler and improved control performance.

The simulation setup with the three first-order processes was used to investigate the influence of  $T_{fbs}$  on the control performance. The results are shown in Fig. 7, which shows the accumulated cost relative to the static case, i.e., using fixed periods. To quantify the impact of computational overhead of the feedback scheduler on the performance of the control loops, the cost has been computed for some different values of  $C_{fbs}$ , the execution time of the feedback scheduler.

As expected, for  $C_{fbs} = 0$ , the cost is decreasing monotonically for decreasing  $T_{fbs}$ . For  $C_{fbs} = 0.05$ , the minimum cost is obtained at  $T_{fbs} = 1.6$ , which corresponds to a feedback scheduler utilization of 3.1 percent. For  $C_{fbs} = 0.1$ , which is the same execution time as for the control tasks, the optimum is located at  $T_{fbs} = 3.5$ , corresponding to a utilization of 2.9 percent. Finally, for  $C_{fbs} = 0.2$ , the optimum period is  $T_{fbs} = 7.6$  (utilization 2.6 percent). It can be seen that the relative performance gain decreases as the execution time of the feedback scheduler increases. For  $C_{fbs} = 0.05$ , the performance gain is 7 percent compared to the static case, for  $C_{fbs} = 0.1$  the gain is down to 3.5 percent, and for  $C_{fbs} = 0.2$  the gain is almost zero.

As seen from the simulation results, the optimal choice of  $T_{fbs}$  depends on the execution time of the feedback scheduler in relation to the other tasks.

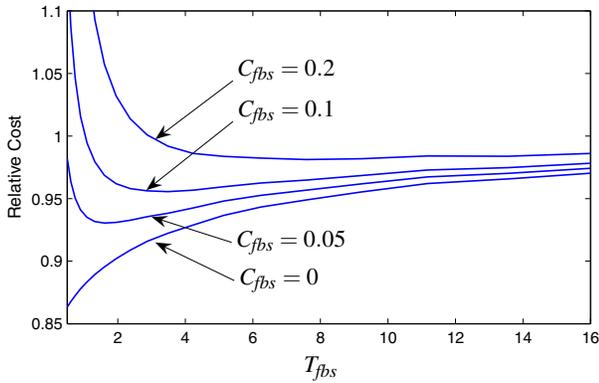


Fig. 7. Relative cost as a function of the feedback scheduler period  $T_{fbs}$  for different values of the feedback scheduler execution time  $C_{fbs}$ .

An alternative approach that would allow for quick response to disturbances while still inducing reasonable overhead, could be to use feedforward from the controller tasks to the scheduler. The tasks would then inform the scheduler when their state exceeded a certain threshold, forcing scheduling actions to be taken.

### B. Computational Delay

In the analysis performed in Sections III and IV we assumed plant models without delay. However, this is unrealistic in a real implementation. By including a fixed input-output latency in the process descriptions and enforcing this in the implementation, the effect of the delay can be incorporated in the cost function.

For instance, for the delayed integrator process

$$dx(t) = u(t - L)dt + dv_c(t), \quad (31)$$

we introduce the extended state vector  $x_e = [x(k) \ u(k-1)]^T$  and get

$$J(x_0, h, T_{fbs}) = x_0^T \bar{S} x_0 + T_{fbs} \left( \frac{\sqrt{3} + 3}{6} h + L \right), \quad (32)$$

where

$$\bar{S} = \begin{bmatrix} L + \frac{\sqrt{3}h}{6} & \frac{L}{6}(3L - \sqrt{3}h) \\ \frac{L}{6}(3L - \sqrt{3}h) & \frac{L^3}{3} - \frac{\sqrt{3}L^2h}{6} \end{bmatrix}. \quad (33)$$

For a given value of  $L$ , we can compute the new expressions relating the cost to the sampling interval and use these in the optimization. To obtain the desired input-output latency in the implementation it is necessary to have predictable sampling and actuation. This can be achieved by the use of a dedicated high-priority task that samples the plant output and actuates the control signal from the last sample at the correct time instants without jitter.

## VI. CONCLUSION

This paper has presented a feedback scheduling strategy for dynamic sampling rate adjustments for a set of LQ-controller tasks. Given the current states of the controlled plants, the sampling periods were chosen to optimize the aggregate cost over the scheduler period. Analytic expressions for the cost as a function of the sampling period and the process state were developed and used as basis for the optimization. The strategy was evaluated by co-simulation of controllers, plants, and the underlying real-time operating system, and performance improvements compared to static sampling rate assignment were illustrated.

In this paper, the performance was measured using quadratic cost functions, and the manipulated computing resource was the sampling periods of the controller tasks. Other performance metrics and scheduling variables may be considered, e.g., [5] considers manipulation of execution times to optimize the performance of a set of model predictive control tasks. Future work will focus on combining these results within a unified feedback scheduling framework.

## REFERENCES

- [1] K. J. Åström and B. Wittenmark, *Computer-Controlled Systems*. Prentice Hall, 1997.
- [2] J. W. S. Liu, *Real-Time Systems*. Prentice Hall, 2000.
- [3] K.-E. Årzén, B. Bernhardsson, J. Eker, A. Cervin, K. Nilsson, P. Persson, and L. Sha, "Integrated control and scheduling," Department of Automatic Control, Lund Institute of Technology, Sweden, Tech. Rep. ISRN LUTFD2/TFRT--7586--SE, Aug. 1999.
- [4] A. Cervin, J. Eker, B. Bernhardsson, and K.-E. Årzén, "Feedback-feedforward scheduling of control tasks," *Real-Time Systems*, vol. 23, no. 1-2, pp. 25-53, July 2002.
- [5] D. Henriksson and J. Åkesson, "Flexible implementation of model predictive control using sub-optimal solutions," Department of Automatic Control, Lund Institute of Technology, Sweden, Tech. Rep. ISRN LUTFD2/TFRT--7610--SE, Apr. 2004.
- [6] D. Seto, J. P. Lehoczky, L. Sha, and K. G. Shin, "On task schedulability in real-time control systems," in *Proc. 17th IEEE Real-Time Systems Symposium*, 1996.
- [7] J. Eker, P. Hagander, and K.-E. Årzén, "A feedback scheduler for real-time control tasks," *Control Engineering Practice*, vol. 8, no. 12, pp. 1369-1378, 2000.
- [8] P. Martí, C. Lin, S. A. Brandt, M. Velasco, and J. M. Fuertes, "Optimal state feedback based resource allocation for resource-constrained control tasks," in *Proc. 23rd IEEE Real-Time Systems Symposium*, 2004.
- [9] B. Lincoln and A. Rantzer, "Optimizing linear system switching," in *Proceedings of the 40th Conference on Decision and Control*, 2001.
- [10] K. J. Åström and B. Bernhardsson, "Comparison of periodic and event based sampling for first-order stochastic systems," in *Preprints 14th World Congress of IFAC*, vol. J, Beijing, P.R. China, July 1999, pp. 301-306.
- [11] M. Schinkel, W.-H. Chen, and A. Rantzer, "Optimal control for systems with varying sampling rate," in *Proceedings of American Control Conference*, Anchorage, May 2002.
- [12] J. A. Stankovic, C. Lu, S. H. Son, and G. Tao, "The case for feedback control real-time scheduling," in *Proc. 11th Euromicro Conference on Real-Time Systems*, 1999.
- [13] K. J. Åström, *Introduction to Stochastic Control Theory*. New York: Academic Press, 1970.
- [14] A. Cervin, D. Henriksson, B. Lincoln, J. Eker, and K.-E. Årzén, "How does control timing affect performance?" *IEEE Control Systems Magazine*, vol. 23, no. 3, pp. 16-30, June 2003.