# Bandwidth-Efficient Controller–Server Co-Design with Stability Guarantees

Amir Aminifar*, Enrico Bini†, Petru Eles*, Zebo Peng*

*Linköping University, Sweden
†Lund University, Sweden

*Abstract*—**Many cyber-physical systems comprise several control applications implemented on a shared platform, for which stability is a fundamental requirement. This is as opposed to the classical hard real-time systems where often the criterion is meeting the deadline. However, the stability of control applications depends on not only the delay experienced, but also the jitter. Therefore, the notion of deadline is considered to be artificial for control applications that promotes the need for new techniques for designing cyber-physical systems. The approach in this paper is built on a server-based resource reservation mechanism, which provides compositionality, isolation, and the opportunity of systematic controller–server co-design. We address the controller–server co-design of such systems to obtain design solutions with the minimal bandwidth to guarantee stability.**

## I. INTRODUCTION AND RELATED WORK

The majority of control applications in cyber-physical systems are implemented by software tasks on processors sharing the available computation resources. The design of such systems requires special attention due to the complex timing behaviors [1] that may, in the worst case, jeopardize the stability of the control applications. These complex timing behaviors are discussed in literature and it is now possible to express control performance and stability in terms of the delay and jitter a control application experiences [2, 3]. The delay and jitter are the direct results of competition for the shared computing resources, i.e., task scheduling.

Typically, in the control–scheduling area [4–13], the tasks are all analyzed or designed together with respect to an overall cost. In such approaches, the design of each control task is tightly coupled with all other control tasks, and hence the design highly suffers from inflexibility since adding, removing, or modifying a task affects all other tasks in the system. In contrast to these approaches, in this paper, we propose to run each controller within a dedicated server in order to achieve isolation for each control task in the execution environment (see Figure 1). The usage of servers presents the following advantages: (1) compositionality that facilitates systematic design methodologies; (2) protection against all *other* tasks' misbehaviors, which may jeopardize the entire system; (3) provides simple interface for systematic controller–server co-design.

In our previous work [14], the problem of analysis and design of stabilizing control servers is addressed, where the controller is considered to be given. In other words, the server is designed to stabilize the plant associated with *a given controller* while consuming the minimum bandwidth.
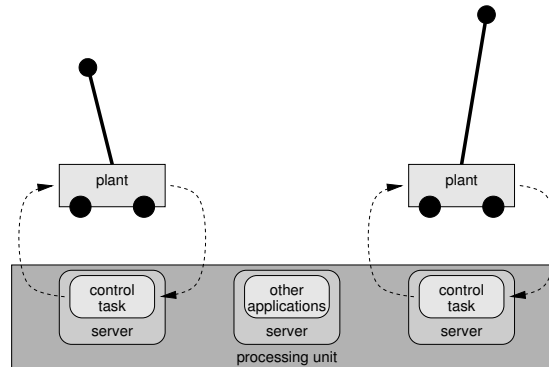
Figure 1. Overview of the proposed approach.

In this paper, we extend our work to address *the controller–server co-design problem*. This co-design process revolutionizes the traditional design flow comprising of two, in fact, dependent design processes (controller and server design) treated, however, independently. Often, the main reason for this separation is the high complexity of each of the design processes as well as the fact that the search space for the co-design problem is the Cartesian product of the search spaces of each of the processes. Therefore, the challenge in approaching this co-design problem is to cope with the complex non-convex objective and constraints, in the huge search space. It is of great importance to realize that the controller and server design are mutually dependent, i.e., a suboptimal design solution is obtained if one of the design processes is performed considering a given, fixed, solution for the other process.

The analysis and design of real-time servers have been extensively studied over the past decade [15–21]. However, the overwhelming majority of the work consider the task deadlines as constraints rather than the stability of control applications. More relevant to this work, Cervin and Eker [22] proposed the control server approach which provides a simple interface used for control-scheduling co-design of real-time systems. Recently, Fontanelli et. al. [23] addressed the problem of optimal bandwidth allocation for a given set of control tasks, restricted to time-triggered models.

In this work, we address the problem of controller–server co-design, for the first time to the best of our knowledge, where the controller is also determined in a unified design process along with the server.

## II. SYSTEM MODEL

A system is composed of $n$ plants, each of which is controlled by a control task that is executing within a server, as shown in Figure 1.

### A. Plant Model

A plant is modeled by a continuous-time system of differential equations [2],

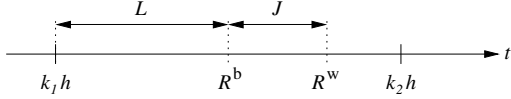$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= Cx, \end{aligned} \tag{1}$$

Figure 2. Graphical interpretation of the nominal delay and worst-case response-time jitter.

where $x$, $u$, and $y$ are the plant state, the control signal, and the plant output, respectively. The index $i$ identifying the plant and the corresponding controller is dropped since each plant is considered in isolation.

### B. Control Task Model

The plant output $y$ is sampled periodically and is used by the controller task $\tau$ to compute the control signal $u$. This control signal will then be applied once the controller completes its execution, at an instant that depends on the scheduling policy and task parameters: the *best-case execution time* ($c^{b}$), the *worst-case execution time* ($c^{w}$), and the *sampling period* ($h$). The sampling frequency is defined to be $f = \frac{1}{h}$.

In addition, the scheduling policy and task parameters also determine the following task characteristics (Figure 2): the *nominal delay* (or latency), denoted by $L = R^{b}$, and the *worst-case response-time jitter* (jitter), denoted by $J = R^{w} - R^{b}$. Note that $R^{w}$ and $R^{b}$ denote the worst-case and best-case response times of the control task, respectively.

### C. Server Model

In this paper, the *periodic server* model [16–19] is considered in which each server $S$ is described by: the *server budget* $Q$, and the *server period* $P$. In this model, the server is activated periodically with period $P$ and in each activation, it allocates $Q$ amount of time to the control task associated with it, before the server deadline expires (server deadline $D = P$). As discussed before, to isolate controllers from one another, each control task is associated with a dedicated server.

As mentioned above, the delay and jitter experienced by a control task are strongly related to the best-case and worst-case response times. The convenient response time upper bound, $\overline{R}^{w}$, and lower bound, $\underline{R}^{b}$, used are given by [14],

$$\overline{R}^{w} = \frac{c^{w}}{\alpha} + \Delta, \qquad \underline{R}^{b} = \max\left\{c^{b}, \frac{c^{b}}{\alpha} - \Delta\right\}, \quad (2)$$

based on Feng–Mok's notation [15],

$$\alpha = \frac{Q}{P}, \qquad \Delta = 2(P - Q), \quad (3)$$

where $\alpha$ and $\Delta$ are the server bandwidth and delay, respectively. Note that the worst-case response time is finite only if no overload occurs [14], that is

$$\alpha = \frac{Q}{P} \geq \frac{c^{w}}{h} = c^{w}f. \quad (4)$$

### III. STABILITY CONSTRAINT

Sufficient conditions to guarantee the stability of a plant (or certain degree of robustness) in presence of delay and jitter in the actuation instant are developed in [3, 24]. The Jitter Margin toolbox [3, 6, 24] computes the stability curve that determines the maximum tolerable response-time jitter $J$ for a nominal delay $L$. The nominal delay $L = R^{b}$
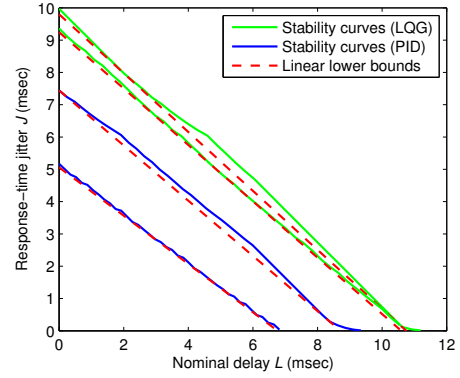


Figure 3. The stability curves generated by Jitter Margin and their linear lower bounds (the area below the curves is the stable area).

identifies the constant part of the delay that the control application experiences, whereas the worst-case response-time jitter $J = R^{w} - R^{b}$ captures the varying part of the delay (see Figure 2).

The solid curves in Figure 3 are examples of the stability curves generated by the Jitter Margin toolbox. The area below the solid curve is the stable area. The green curves are generated for a DC servo process $\frac{1000}{s^{2}+s}$ and discrete-time Linear-Quadratic-Gaussian (LQG) controllers. The upper and lower green curves correspond to LQG controllers with sampling periods 6 $ms$ and 12 $ms$, respectively. Replacing the Linear-Quadratic-Gaussian controllers with a Proportional-Integral-Derivative (PID) controller, the blue curves in Figure 3 are obtained. Notice the large gap between the two blue curves generated for the two sampling periods. The upper blue curve, $h = 6$ $ms$, provides better stability margin (the application can tolerate larger delay and jitter) compared to the lower curve, $h = 12$ $ms$, but requires higher bandwidth due to more frequent activation. This trade-off motivates the need for a co-design approach where the real-time parameters are optimized along with the controllers in a unified process.

For a given sampling period, the stability curve can safely be approximated by a linear function of the nominal delay and worst-case response-time jitter. The linear *stability constraint* can be formulated as,

$$L + aJ \leq b, \quad (5)$$

where $a \geq 1, b \geq 0$. The linear lower bounds, depicted by the dashed lines, on the original curves generated by the Jitter Margin toolbox are also shown in Figure 3.

In the previous design approach [14], the stability constraint is captured by constant coefficients $a$ and $b$, which means that we consider the controller and its period to be fixed and given. In this paper, we assume that the controller will be determined as part of the co-design process. Therefore, the variables $a$ and $b$ are functions of the sampling period $h$ (or alternatively sampling frequency $f$) of the controller. Let us assume that the coefficients $a$ and $b$ are expressed as polynomial functions of the sampling frequency $f$. These polynomials are obtained considering the set of curves each of which generated with an optimal controller synthesized for the given sampling frequency and a constant delay. Observe that, the stability region with the approximated coefficients must always be contained within the original one generated by the Jitter Margin toolbox.

This can be enforced by, e.g., using the approximated $a$ and $b$ polynomials that are an upper bound and a lower bound of their original exact functions, respectively.

In the context of the optimization problem that we are solving, the safe upper/lower bounds of the worst/best-case response times of (2), are used instead of the exact worst-case and best-case response times, since the discontinuity of the exact response times makes the optimization problem impractical. Hence, the nominal delay and the worst-case response-time jitter are redefined based on the bounds in Section II-C,

$$\begin{aligned} \underline{L} &= \underline{R}^{\mathrm{b}}, \\ \overline{J} &= \overline{R}^{\mathrm{w}} - \underline{R}^{\mathrm{b}}. \end{aligned} \tag{6}$$

While using the bounds in equation (2) involves some pessimism compared to the original supply bounds, their usage is safe from the stability point of view [6].

The stability constraint based on the bounds is given in the following,

$$\begin{aligned} b &\geq \underline{L} + a\overline{J} = \underline{R}^{\mathrm{b}} + a(\overline{R}^{\mathrm{w}} - \underline{R}^{\mathrm{b}}) \\ &= a\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta\right) - (a-1)\max\left\{c^{\mathrm{b}}, \frac{c^{\mathrm{b}}}{\alpha} - \Delta\right\}, \end{aligned}$$

which can be rewritten as

$$\min\left\{a\left(\frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta\right) + \frac{c^{\mathrm{b}}}{\alpha} - \Delta - b,\right.$$
$$\left. a\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) + c^{\mathrm{b}} - b\right\} \leq 0. \tag{7}$$

Hence, equation (7) captures the constraint on the server parameters, $\alpha$ and $\Delta$, as well as the sampling frequency $f$ of the controller in order to guarantee stability. Note that $a$ and $b$ are functions of the sampling frequency $f$, although the sampling frequency $f$ does not explicitly appear in (7).

The degree of pessimism introduced in the analysis by using the bounds in equation (7) instead of the exact response times is discussed in [14]. It is shown that if a server $S_1 = (Q, P)$ (with the exact expressions for the response time) with bandwidth $\alpha_1 = \frac{Q}{P}$ is identified that guarantees the stability of the control task associated with it, then there exists a server $S_2 = (\frac{Q}{2}, \frac{P}{2})$ (with the bounds in equation (2)) that can also guarantee the stability of the control task and the required bandwidth is *the same*, i.e., $\alpha_2 = \frac{\frac{Q}{2}}{\frac{P}{2}} = \frac{Q}{P}$. This result also states that, *in the worst-case*, the server $S_2$, while consuming the same bandwidth, has to be run two times more frequently compared to $S_1$, which might be a disadvantage, if the context-switch overhead is significant.

## IV. MOTIVATIONAL EXAMPLE

To motivate the co-design optimization problem, a small example with three plants is considered in this section. The real-time parameters of the control tasks are given in Table I. All timing quantities are given in units of $0.1\ ms$ throughout this section. The interval for the sampling period, i.e., $\left[h_i^{\min}, h_i^{\max}\right]$, is obtained based on the common rules of thumb that state that the sampling period can be chosen in the interval of $\left[\frac{0.2}{\omega_b}, \frac{0.6}{\omega_b}\right]$, with $\omega_b$ being the bandwidth of the closed-loop continuous system [2]. The coefficients $a$ and $b$ are captured by polynomials of degree

### TABLE I
MOTIVATIONAL EXAMPLE

| $i$ | Parameters | | | | SD Approach | | CD Approach | | |
|---|---|---|---|---|---|---|---|---|---|
| | $c_i^{\mathrm{b}}$ | $c_i^{\mathrm{w}}$ | $h_i^{\min}$ | $h_i^{\max}$ | $\alpha_i^*$ | $\Delta_i^*$ | $h_i^*$ | $\alpha_i^*$ | $\Delta_i^*$ |
| 1 | 5 | 10 | 46 | 138 | .2 | 5.7 | 69 | .14 | 9.9 |
| 2 | 20 | 40 | 101 | 303 | .32 | 6.3 | 163 | .25 | 12.5 |
| 3 | 29 | 58 | 142 | 427 | .33 | 7.3 | 231 | .25 | 14.9 |

5 based on the sampling frequency $f$.[1] For the sake of simplicity, in this example, we assume that the continuous-time controllers are given, but their sampling periods are design parameters.

Let us first consider the case in which the three sampling periods $h_1 = 92$, $h_2 = 202$, and $h_3 = 285$ are given ($h_i = \frac{h_i^{\min} + h_i^{\max}}{2}$). The problem as formulated in the server design (SD) approach [14] is to find the optimal parameters $\alpha$ and $\Delta$ for the servers such that the total utilization is minimized and worst-case stability is guaranteed. The solution is presented in Table I column group 2 and the total utilization is $92\%$.

The co-design approach (CD) presented in this paper goes further. The goal is to obtain the optimal sampling periods and server parameters such that the total utilization is minimized while guaranteeing stability. The resulting controller periods and server parameters are shown in Table I column group 3 with a total utilization of only $68\%$.

It is interesting to observe that, if we apply the server design approach to optimize the server for given controllers corresponding to the shortest $h^{\min}$ and longest $h^{\max}$ periods, the resulting utilizations are $102\%$ and $239\%$, respectively. This means, first of all, that none of these alternatives can be implemented such that stability is guaranteed. It also shows that the utilization, considering the longest sampling period $h^{\max}$ can (somewhat counter-intuitively) be much larger than the one with the smallest sampling period $h^{\min}$. This is due to the fact that, in order to compensate for the large sampling period and still approach stability, the server parameters (period and budget) have to be fixed such that delay and jitter are reduced. This, finally, leads to the increased utilization. This intricate relation among controller parameters, server parameters, and stability condition leads to the increased complexity of the problem and the need for an efficient controller–server co-design process.

## V. PROBLEM FORMULATION

The inputs of our optimization problem are,
- the best-case and worst-case execution times, denoted by $c^{\mathrm{b}}$ and $c^{\mathrm{w}}$, respectively, and
- the $a$ and $b$ polynomial functions of the sampling frequency $f$, valid in the allowed sampling frequency interval $\left[f^{\min}, f^{\max}\right]$.

The outputs are the server bandwidth $\alpha$, and delay $\Delta$, as well as the controller. The optimization problem identifies also a unique optimal sampling frequency, $f = \frac{1}{h}$, that corresponds to an optimal controller.

---

[1]Assuming the sampling period and the controller to be fixed, the safe and constant coefficients $a$ and $b$ are obtained by a constrained least-square optimization, which is convex. In this paper, however, $a$ and $b$ are functions of sampling frequency. To find these polynomials, first, we compute the constant $a$ and $b$ coefficients, as in our previous work, for several stability curves generated for different sampling frequencies and corresponding optimal LQG controllers in the interval identified by the rules of thumb. Having these constant $a$ and $b$ coefficients for different sampling frequencies, we solve two convex constrained least-square optimizations to find the polynomials $a$ and $b$.

The objective of the optimization is to minimize the utilization required to guarantee the stability of all controllers. That is, our goal is to minimize

$$U = \sum_{i=1}^{n} \left( \alpha_i + \frac{\epsilon}{P_i} \right), \tag{8}$$

under the stability constraint (7) and the validity constraint for the response time bounds (4), for each closed-loop system. The *strictly* positive $\epsilon$ represents the switching overhead of the servers.

The controller–server co-design for each plant can be solved independently of all others as a result of the isolation property under the resource reservation mechanism. Hence, instead of dealing with an optimization involving all plants, the minimization of the total server utilization of (8) is divided into one bandwidth minimization problem for each plant in our approach. Thus, the following optimization for each plant is performed,

$$\min_{\alpha,\Delta,f} \quad \alpha + \frac{2\epsilon(1-\alpha)}{\Delta}$$
$$\text{s.t.} \quad \min \left\{ a \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) + \frac{c^{\mathrm{b}}}{\alpha} - \Delta - b, \right. \tag{9}$$
$$\left. a \left( \frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}} \right) + c^{\mathrm{b}} - b \right\} \leq 0,$$
$$c^{\mathrm{w}} f - \alpha \leq 0,$$

where the period $P$ is replaced by $\frac{\Delta}{2(1-\alpha)}$, based on (3). Observe that $a$ and $b$ are polynomial functions of the sampling frequency $f$. The first constraint guarantees the stability of the control applications (see constraint (7)), while the second one ensures that server is never overloaded and then the response time is always finite (see constraint (4)).

## VI. Controller–Server Co-Design

In this section, the procedure of controller–server co-design is discussed. First, observe that the stability constraint in (9) can be written as

$$\min \{ g_{\mathrm{I}}(\alpha, \Delta, f), g_{\mathrm{II}}(\alpha, \Delta, f) \} \leq 0,$$

which is equivalent to

$$(g_{\mathrm{I}}(\alpha, \Delta, f) \leq 0) \quad \vee \quad (g_{\mathrm{II}}(\alpha, \Delta, f) \leq 0),$$

with $\vee$ denoting the *logical or* between the two propositions. In other words, solving the problem (9) is equivalent to solving the following two problems,

$$\min_{\alpha,\Delta,f} \quad \alpha + \frac{2\epsilon(1-\alpha)}{\Delta}$$
$$\text{s.t.} \quad a \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) + \frac{c^{\mathrm{b}}}{\alpha} - \Delta - b \leq 0, \tag{10}$$
$$c^{\mathrm{w}} f - \alpha \leq 0,$$

and,

$$\min_{\alpha,\Delta,f} \quad \alpha + \frac{2\epsilon(1-\alpha)}{\Delta}$$
$$\text{s.t.} \quad a \left( \frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}} \right) + c^{\mathrm{b}} - b \leq 0, \tag{11}$$
$$c^{\mathrm{w}} f - \alpha \leq 0,$$

and then selecting the best solution produced by (10) and (11).

To solve problems (10) and (11), we use the KKT (Karush-Kuhn-Tucker) necessary conditions for optimality [25]. According to the KKT condition, the optimum $\boldsymbol{x}^*$ of the problem

$$\min_{\boldsymbol{x}} \quad f(\boldsymbol{x})$$
$$\text{s.t.} \quad g_i(\boldsymbol{x}) \leq 0, \qquad i = 1 \ldots m, \tag{12}$$

must necessarily satisfy the following conditions

$$\nabla f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \mu_i^* \nabla g_i(\boldsymbol{x}^*) = \mathbf{0},$$
$$\mu_i^* g_i(\boldsymbol{x}^*) = 0, \qquad i = 1 \ldots m, \tag{13}$$
$$\mu_i^* \geq 0, \qquad i = 1 \ldots m.$$

### Solving problem (10):

From the KKT condition of the gradient (the first equality), if we differentiate w.r.t. $\alpha$, $\Delta$, and $f$, respectively, we find[2]

$$1 - \frac{2\epsilon}{\Delta} - \mu_1 \frac{a(c^{\mathrm{w}} - c^{\mathrm{b}}) + c^{\mathrm{b}}}{\alpha^2} - \mu_2 = 0 \tag{14}$$
$$-\frac{2\epsilon(1-\alpha)}{\Delta^2} + \mu_1(2a - 1) = 0 \tag{15}$$
$$\mu_1 \left( \dot{a} \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) - \dot{b} \right) + \mu_2 c^{\mathrm{w}} = 0 \tag{16}$$

Using the second equality in (13), we obtain two more equations as follows,

$$\mu_1 \left( a \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) + \frac{c^{\mathrm{b}}}{\alpha} - \Delta - b \right) = 0 \tag{17}$$
$$\mu_2 (c^{\mathrm{w}} f - \alpha) = 0 \tag{18}$$

Since $a \geq 1$ and $\alpha < 1$, from (15), we immediately find the multiplier $\mu_1$, that is:

$$\mu_1 = \frac{2\epsilon(1-\alpha)}{\Delta^2(2a - 1)} > 0, \tag{19}$$

hence the stability constraint of (10) is active and must hold with the equal sign, i.e.,

$$a \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) + \frac{c^{\mathrm{b}}}{\alpha} - \Delta - b = 0. \tag{20}$$

Using the above equation, $\Delta$ is given as follows,

$$\Delta = \frac{b - \frac{a(c^{\mathrm{w}} - c^{\mathrm{b}}) + c^{\mathrm{b}}}{\alpha}}{2a - 1}. \tag{21}$$

From (16), we obtain,

$$\mu_2 = \frac{-\mu_1}{c^{\mathrm{w}}} \left( \dot{a} \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) - \dot{b} \right). \tag{22}$$

Considering equality (18) and substituting $\mu_2$ from the above, we obtain,

$$\frac{-\mu_1}{c^{\mathrm{w}}} \left( \dot{a} \left( \frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta \right) - \dot{b} \right) (c^{\mathrm{w}} f - \alpha) = 0. \tag{23}$$

We consider two cases: $\alpha = c^{\mathrm{w}} f$ and $\alpha > c^{\mathrm{w}} f$. Let us first assume $\alpha = c^w f$ that implies the equality above is satisfied. From (21), the corresponding server delay is

[2] $\dot{a} = \frac{\partial a}{\partial f}, \quad \dot{b} = \frac{\partial b}{\partial f}.$

$\Delta = \frac{bc^{\mathrm{w}}f - a(c^{\mathrm{w}} - c^{\mathrm{b}}) - c^{\mathrm{b}}}{(2a-1)c^{\mathrm{w}}f}$. From equality (14), (19), and (22), we obtain the following equation based on $\alpha$ and $\Delta$,

$$1 - \frac{2\epsilon}{\Delta} - \frac{2\epsilon(1-\alpha)}{\Delta^2(2a-1)}\left(\frac{a(c^{\mathrm{w}} - c^{\mathrm{b}}) + c^{\mathrm{b}}}{\alpha^2}\right.$$
$$\left. -\frac{1}{c^{\mathrm{w}}}\left(\dot{a}\left(\frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta\right) - \dot{b}\right)\right) = 0. \quad (24)$$

Using $\alpha = c^{\mathrm{w}}f$ and $\Delta = \frac{bc^{\mathrm{w}}f - a(c^{\mathrm{w}} - c^{\mathrm{b}}) - c^{\mathrm{b}}}{(2a-1)c^{\mathrm{w}}f}$ in equation (24) and solving it for frequency $f$, the optimal sampling frequency candidates will be obtained. Corresponding $\alpha$ and $\Delta$ are given based on the equations discussed. Let us denote this set of design solutions by $\mathrm{DS}_1$.

If instead $\alpha > c^{\mathrm{w}}f$, the multiplier $\mu_1$ is strictly positive and therefore, equality (23) holds iff,

$$\dot{a}\left(\frac{c^{\mathrm{w}} - c^{\mathrm{b}}}{\alpha} + 2\Delta\right) - \dot{b} = 0, \quad (25)$$

Observe that the equality corresponds to $\mu_2 = 0$, according to (18) and taking into consideration that we have $\alpha > c^{\mathrm{w}}f$. Therefore, from (14) and (19) we obtain,

$$1 - \frac{2\epsilon}{\Delta} - \frac{2\epsilon(1-\alpha)}{\Delta^2(2a-1)}\left(\frac{a(c^{\mathrm{w}} - c^{\mathrm{b}}) + c^{\mathrm{b}}}{\alpha^2}\right) = 0. \quad (26)$$

Substituting (21) in equality (25), the server bandwidth is given by

$$\alpha = \frac{(c^{\mathrm{w}} + c^{\mathrm{b}})\dot{a}}{2b\dot{a} - \dot{b}(2a-1)} = \frac{(c^{\mathrm{w}} + c^{\mathrm{b}})\dot{a}}{\dot{b} - 2(a\dot{b} - b\dot{a})}. \quad (27)$$

Substituting back $\alpha$ in (21) we have,

$$\Delta = \frac{\dot{b}c^{\mathrm{b}} + (a\dot{b} - b\dot{a})(c^{\mathrm{w}} - c^{\mathrm{b}})}{(c^{\mathrm{w}} + c^{\mathrm{b}})\dot{a}}. \quad (28)$$

Using $\alpha$ and $\Delta$ in equation (26) and solving the polynomial based on sampling frequency $f$, the candidates for the optimal sampling frequency are identified. The corresponding $\alpha$ and $\Delta$ are obtained from the above equations. The set of solutions obtained for this case are captured by $\mathrm{DS}_2$.

***Solving problem (11):***

Analogously, problem (11) is solved based on the KKT conditions,

$$1 - \frac{2\epsilon}{\Delta} - \mu_1\frac{ac^{\mathrm{w}}}{\alpha^2} - \mu_2 = 0 \quad (29)$$

$$-\frac{2\epsilon(1-\alpha)}{\Delta^2} + \mu_1 a = 0 \quad (30)$$

$$\mu_1\left(\dot{a}\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) - \dot{b}\right) + \mu_2 c^{\mathrm{w}} = 0 \quad (31)$$

The second equality in (13) leads to two more equations,

$$\mu_1\left(a\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) + c^{\mathrm{b}} - b\right) = 0 \quad (32)$$

$$\mu_2\left(c^{\mathrm{w}}f - \alpha\right) = 0 \quad (33)$$

Since $a \geq 1$ and $\alpha < 1$, we conclude that,

$$\mu_1 = \frac{2\epsilon(1-\alpha)}{\Delta^2 a} > 0, \quad (34)$$

which, considering (32), implies,

$$a\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) + c^{\mathrm{b}} - b = 0. \quad (35)$$

Then, the server delay is obtain as follows,

$$\Delta = \frac{b + (a-1)c^{\mathrm{b}}}{a} - \frac{c^{\mathrm{w}}}{\alpha}. \quad (36)$$

From (31), we find $\mu_2$ as given by,

$$\mu_2 = \frac{-\mu_1}{c^{\mathrm{w}}}\left(\dot{a}\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) - \dot{b}\right), \quad (37)$$

and from (33), the following equation is obtained,

$$\frac{-\mu_1}{c^{\mathrm{w}}}\left(\dot{a}\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) - \dot{b}\right)(c^{\mathrm{w}}f - \alpha) = 0. \quad (38)$$

Similar to the approach used for problem (10), we consider two cases: $\alpha = c^{\mathrm{w}}f$ and $\alpha > c^{\mathrm{w}}f$. First, let us assume $\alpha = c^{\mathrm{w}}f$ that leads to the corresponding $\Delta = \frac{b+(a-1)c^{\mathrm{b}}}{a} - \frac{1}{f}$. Using (29), (34), (37), and (35), we obtain,

$$1 - \frac{2\epsilon}{\Delta} - \frac{2\epsilon(1-\alpha)}{\Delta^2 a}\left(\frac{ac^{\mathrm{w}}}{\alpha^2} - \frac{\left(\dot{a}\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) - \dot{b}\right)}{c^{\mathrm{w}}}\right) =$$
$$1 - \frac{2\epsilon}{\Delta} - \frac{2\epsilon(1-\alpha)}{\Delta^2 a}\left(\frac{ac^{\mathrm{w}}}{\alpha^2} - \frac{1}{c^{\mathrm{w}}}\left(\dot{a}\frac{b - c^{\mathrm{b}}}{a} - \dot{b}\right)\right) = 0.$$

Substituting $\alpha = c^{\mathrm{w}}f$ and $\Delta = \frac{b+(a-1)c^{\mathrm{b}}}{a} - \frac{1}{f}$ above, we end up with an equation based on $f$ in which the roots are the candidates for the optimal sampling frequency. Corresponding $\alpha$ and $\Delta$ are calculated based on the equations discussed. We denote these design solutions by $\mathrm{DS}_3$.

Let us now consider the case with $\alpha > c^{\mathrm{w}}f$. Since $\mu_1$ is strictly positive, the equality (38) holds iff,

$$\dot{a}\left(\frac{c^{\mathrm{w}}}{\alpha} + \Delta - c^{\mathrm{b}}\right) - \dot{b} = 0, \quad (39)$$

Considering (35) and (39), we obtain an equation for the candidates of the optimal sampling frequency as follows,

$$\dot{a}\frac{b - c^{\mathrm{b}}}{a} - \dot{b} = 0.$$

Note that in this case, we know $\mu_2 = 0$ (from (33) and knowing $\alpha > c^{\mathrm{w}}f$), and therefore based on (29), (34), and $\Delta$ in (36), we have a quadratic equation for $\alpha$ based on each candidate $f$,

$$\left(\left(\frac{b + (a-1)c^{\mathrm{b}}}{a}\right)^2 - 2\epsilon\frac{b + (a-1)c^{\mathrm{b}}}{a}\right)\alpha^2 +$$
$$\left(-2c^{\mathrm{w}}\frac{b + (a-1)c^{\mathrm{b}}}{a}\right)\alpha + (c^{\mathrm{w}2} - 2\epsilon c^{\mathrm{w}}) = 0.$$

Substituting back $\alpha$ in (36), $\Delta$ is found. The corresponding design solutions are denoted by $\mathrm{DS}_4$.

We have obtained four sets of tuples $(\alpha, \Delta, f)$ as candidates for the optimal co-design solution. The best solution obtained is the one that results in the least total utilization and is valid,[3]

$$\min_{\bigcup_{i=1}^{5}\mathrm{DS}_i}\left\{\alpha + \frac{2\epsilon(1-\alpha)}{\Delta} \,\middle|\, \frac{c^{\mathrm{w}}}{h} \leq \alpha < 1, \ \Delta \geq 2\epsilon, \ f^{\min} \leq f \leq f^{\max}\right\}.$$

[3]In addition, the boundary values of sampling frequency, i.e., $f^{\min}$ and $f^{\max}$, need to be considered among the solutions. The corresponding server bandwidth and delay, for these boundary values, can be obtained based on the previous work [14]. The set $\mathrm{DS}_5$ includes the design solutions for the boundary values.
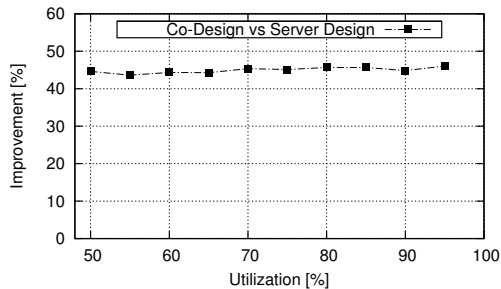
Figure 4. The average improvement in total utilization by the proposed co-design approach in this paper against the approach in [14].

Having found the minimum resource utilization required for stability of all plants, we should now check if the total resource demand is not exceeding the resource supply. The solution found is valid if and only if the total utilization is less than or equal to one, i.e.,

$$\sum_{i=1}^{n} \left( \alpha_i^* + \frac{2\epsilon(1 - \alpha_i^*)}{\Delta_i^*} \right) \leq 1. \quad (40)$$

If inequality (40) is not satisfied, there is no possible implementation to guarantee the worst-case stability of all applications, under the constraint in (7).

## VII. EXPERIMENTAL RESULTS

To evaluate the efficiency of our co-design approach (CD), we compare it against the server design approach (SD) in the previous work [14]. The comparison is performed for a set of 100 benchmarks with 2 to 10 plants from a database consisting of inverted pendulums, DC servos, ball and beam processes, and harmonic oscillators, which are extensively used in literature for experimental evaluation [2, 6]. The random task sets, for a given utilization, are generated by the UUniFast algorithm in [26]. The coefficients $a$ and $b$ are considered to be polynomials of degree 5. The switching overhead is 1–5% of the minimum best-case execution-time. In the server design approach [14], the sampling frequency is considered in the middle of the sampling frequency interval identified by the common rules of thumb [2].

The experiments are repeated for different taskset utilizations and the results are shown in Figure 4. As can be seen, the co-design approach outperforms the previous work by an average of $44\%$. The relative improvement is defined as $\left( \frac{U_{SD} - U_{CD}}{U_{SD}} \times 100 \right)$, where $U_{CD}$ and $U_{SD}$ denote the total utilization, $\sum_{i=1}^{n} \left( \alpha_i^* + \frac{2\epsilon(1 - \alpha_i^*)}{\Delta_i^*} \right)$, consumed by the optimal co-design and server design approaches, respectively.

The time complexity of the proposed co-design approach, for a single application, is in the order of finding the roots of a polynomial. Due to the isolation provided by the resource reservation mechanism, the timing complexity grows linearly with the number of control applications that demonstrates the scalability of our proposed approach.

## VIII. CONCLUSIONS

In this paper, we have addressed the problem of controller–server co-design in a systematic way. It has been shown that the co-design approach outperforms the previous work where the controllers are considered to be given. This result is not surprising since the combined

solution space is much larger for the co-design problem. However, the challenge is the optimization of *complex objectives*, under *complex constraints*, in *the huge search space*, i.e., what made the co-design problem to be treated as two separate design processes in the first place. We have proposed an approach based on the Karush-Kuhn-Tucker optimality conditions and demonstrated its efficiency.

## REFERENCES

[1] Björn Wittenmark et al. "Timing Problems in Real-Time Control Systems". In: *Proceedings of the American Control Conference*. 1995, pp. 2000–2004.

[2] K. J. Åström and B. Wittenmark. *Computer-Controlled Systems*. 3rd ed. Prentice Hall, 1997.

[3] A. Cervin. "Stability and Worst-Case Performance Analysis of Sampled-Data Control Systems with Input and Output Jitter". In: *Proceedings of the 2012 American Control Conference (ACC)*. 2012.

[4] D. Seto et al. "On Task Schedulability in Real-Time Control Systems". In: *Proceedings of the 17th IEEE Real-Time Systems Symposium*. 1996, pp. 13–21.

[5] H. Rehbinder and M. Sanfridson. "Integration of Off-Line Scheduling and Optimal Control". In: *Proceedings of the 12th Euromicro Conference on Real-Time Systems*. 2000, pp. 137–143.

[6] Anton Cervin et al. "The Jitter Margin and Its Application in the Design of Real-Time Control Systems". In: *Proceedings of the 10th International Conference on Real-Time and Embedded Computing Systems and Applications*. 2004.

[7] Truong Nghiem et al. "Time-triggered implementations of dynamic controllers". In: *Proceedings of the 6th ACM & IEEE International conference on Embedded software*. 2006, pp. 2–11.

[8] E. Bini and A. Cervin. "Delay-Aware Period Assignment in Control Systems". In: *Proceedings of the 29th IEEE Real-Time Systems Symposium*. 2008, pp. 291–300.

[9] Fumin Zhang et al. "Task Scheduling for Control Oriented Requirements for Cyber-Physical Systems". In: *Proceedings of the 29th IEEE Real-Time Systems Symposium*. 2008, pp. 47–56.

[10] Payam Naghshtabrizi and João Pedro Hespanha. "Analysis of Distributed Control Systems with Shared Communication and Computation Resources". In: *Proceedings of the 2009 American Control Conference (ACC)*. 2009.

[11] Rupak Majumdar et al. "Performance-aware scheduler synthesis for control systems". In: *Proceedings of the 9th ACM international conference on Embedded software*. 2011, pp. 299–308.

[12] Pratyush Kumar et al. "A Hybrid Approach to Cyber-Physical Systems Verification". In: *Proceedings of the 49th Design Automation Conference*. 2012.

[13] Amir Aminifar et al. "Designing High-Quality Embedded Control Systems with Guaranteed Stability". In: *Proceedings of the 33th IEEE Real-Time Systems Symposium*. 2012, pp. 283–292.

[14] Amir Aminifar et al. "Designing Bandwidth-Efficient Stabilizing Control Servers". In: *Proceedings of the 34th IEEE Real-Time Systems Symposium*. 2013.

[15] Xiang Feng and A.K. Mok. "A model of hierarchical real-time virtual resources". In: *Proceedings of the 23th IEEE Real-Time Systems Symposium*. 2002, pp. 26–35.

[16] S. Saewong et al. "Analysis of hierar hical fixed-priority scheduling". In: *Proceedings of the 14th Euromicro Conference on Real-Time Systems*. 2002, pp. 152–160.

[17] G. Lipari and E. Bini. "Resource partitioning among real-time applications". In: *Proceedings of the 15th Euromicro Conference on Real-Time Systems*. 2003, pp. 151–158.

[18] Insik Shin and Insup Lee. "Periodic Resource Model for Compositional Real-Time Guarantees". In: *Proceedings of the 24th IEEE Real-Time Systems Symposium*. 2003, pp. 2–13.

[19] Luis Almeida and Paulo Pedreiras. "Scheduling within temporal partitions: response-time analysis and server design". In: *Proceedings of the 4th ACM international conference on Embedded software*. 2004, pp. 95–103.

[20] Arvind Easwaran et al. "Compositional analysis framework using EDP resource models". In: *Proceedings of the 28th IEEE Real-Time Systems Symposium*. 2007, pp. 129–138.

[21] Nathan Fisher and Farhana Dewan. "A bandwidth allocation scheme for compositional real-time systems with periodic resources". In: *Real-Time Systems* 48.3 (2012), pp. 223–263.

[22] Anton Cervin and Johan Eker. "Control-scheduling codesign of real-time systems: The control server approach". In: *Journal of Embedded Computing* 1.2 (2005), pp. 209–224.

[23] Daniele Fontantelli et al. "Optimal CPU allocation to a set of control tasks with soft real–time execution constraints". In: *Proceedings of the 16th international conference on Hybrid systems: computation and control*. 2013, pp. 233–242.

[24] Chung-Yao Kao and Bo Lincoln. "Simple stability criteria for systems with time-varying delays". In: *Automatica* 40 (2004), pp. 1429–1434. ISSN: 0005-1098.

[25] M.S. Bazaraa et al. *Nonlinear Programming: Theory and Algorithms*. Wiley, 2006. ISBN: 9780471787761.

[26] Enrico Bini and Giorgio C. Buttazzo. "Measuring the Performance of Schedulability Tests". In: *Real-Time Systems* 30.1-2 (2005), pp. 129–154.