

# Implementing Optimization-Based Control Tasks in Cyber-Physical Systems With Limited Computing Capacity

Mehdi Hosseinzadeh, Bruno Sinopoli, Ilya Kolmanovsky, and Sanjoy Baruah

## I. INTRODUCTION AND MOTIVATION

A common aspect of today's Cyber-Physical Systems (CPSs) is that multiple control tasks may execute in a shared processor. For a CPS performing  $n$  optimization-based control tasks, denoted by  $\tau_k$ ,  $k = 1, \dots, n$ , let  $\ell_k$  and  $\Delta T_k$  denote the worst-case execution time and sampling period of task  $\tau_k$ , respectively. Under the Earliest Deadline First (EDF) policy, the tasks are schedulable on a single preemptive processor if the following condition [1] is satisfied:

$$\sum_{k=1}^n \frac{\ell_k}{\Delta T_k} \leq 1. \quad (1)$$

Optimization-based control tasks make use of online optimization and thus have large execution times; hence their sampling periods must be large as well to satisfy condition (1). However larger sampling periods may cause worse control performance.

**Prior Work.** Existing methods to address the above-mentioned issue are: i) control-scheduling co-design (e.g., [2]), where parameters of control systems are modified at every sampling instant; ii) pre-computation (e.g., [3]), where optimal control inputs are computed offline and stored for run-time use; iii) triggering-based control (e.g., [4]), where a triggering mechanism invokes control tasks; and iv) fixed-iteration optimization (e.g., [5]), where control tasks perform a fixed number of iterations to approximately track the solution. These methods either do not guarantee constraint satisfaction (e.g., [2]), or do not consider the variability and unpredictability of the task execution time (e.g., [3]) and available computing time (e.g., [4], [5]).

Recently, dynamically embedded controllers have been proposed in the control theory literature (e.g., [6], [7]), where the processor, instead of solving an optimization problem, runs a virtual dynamical system whose trajectory converges to the optimal solution. This type of an approach is also pursued in our work limited and variable computing capacity in implementing optimization-based control tasks in CPSs.

**Goal.** Drawing inspiration from dynamically embedded controllers, the goal of our work is to develop a robust to

This work has been supported by National Science Foundation (NSF) under grant numbers ECCS-1931738, ECCS-1932530, and CMMI-1904394.

M. Hosseinzadeh and B. Sinopoli are with the Department of Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, MO 63130, USA (email: mehdi.hosseinzadeh@ieee.org; bsinopoli@wustl.edu). I. Kolmanovsky is with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, USA (email: ilya@umich.edu). S. Baruah is with the Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO 63130, USA (email: baruah@wustl.edu).

early termination optimization approach that can be used to effectively solve onboard optimization problems involved in controlling the system despite the presence of unpredictable, variable, and limited computing capacity.

## II. PROPOSED SOLUTION—ROBUST TO EARLY TERMINATION OPTIMIZATION APPROACH

**Task Details.** Suppose that task  $\tau_k$ ,  $k \in \{1, \dots, n\}$  solves the following optimization problem:

$$\begin{cases} \min_x f_0(x) \\ \text{s.t. } f_i(x) \leq 0, \quad i = 1, \dots, m \end{cases}, \quad (2)$$

where  $f_0 : \mathbb{R}^p \rightarrow \mathbb{R}$  is a strongly convex objective function to be minimized over the  $p$ -variable vector  $x$ , and  $f_i(x) \leq 0$  is the  $i$ -th inequality constraint. Note that the mathematical problems in many existing optimization-based control tasks (e.g., Model Predictive Control (MPC) [8]) are in the form of optimization problem (2).

**Proposed Method.** Consider the following *modified* barrier function associated with the optimization problem (2):

$$\mathcal{B}(x, \lambda) = f_0(x) - \sum_{i=1}^m \lambda_i \log(-\beta(f_i(x) + 1/\beta) + 1), \quad (3)$$

where  $\beta \in \mathbb{R}_{>0}$  is the barrier parameter and  $\lambda = [\lambda_1 \dots \lambda_m]^\top \in \mathbb{R}_{>0}^m$  is the vector of dual parameters.

We consider the following primal-dual gradient flow:

$$\dot{\hat{x}}(t) = -\sigma \nabla_{\hat{x}} \mathcal{B}(\hat{x}(t), \lambda(t)), \quad (4a)$$

$$\dot{\hat{\lambda}}(t) = +\sigma \left( \nabla_{\hat{\lambda}} \mathcal{B}(\hat{x}(t), \lambda(t)) + \Psi_i(t) \right), \quad (4b)$$

where  $\sigma \in \mathbb{R}_{>0}$  is a design parameter and  $\Psi_i(t)$  is the projection operator onto the normal cone of  $\lambda_i$  (see [6]).

**Convergence.** Let  $\beta$  be sufficiently large. Using the following Lyapunov function:

$$V(\hat{x}(t), \hat{\lambda}(t)) = \frac{1}{2\sigma} \|\hat{x}(t) - x^*\|^2 + \frac{1}{2\sigma} \|\hat{\lambda}(t) - \lambda^*\|^2, \quad (5)$$

where  $(x^*, \lambda^*)$  is the pair of the optimal solution of (2) and the vector of optimal dual parameters, and according to the fact that the operator  $\left[ (\nabla_{\hat{x}} \mathcal{B}(\hat{x}, \hat{\lambda}))^\top - (\nabla_{\hat{\lambda}} \mathcal{B}(\hat{x}, \hat{\lambda}))^\top \right]^\top$  is strongly monotone, it can be shown [7] that  $(\hat{x}(t), \hat{\lambda}(t))$  exponentially converges to  $(x^*, \lambda^*)$  as  $t \rightarrow \infty$ .

**Constraint-Handling.** Since  $\mathcal{B}(\hat{x}(t), \hat{\lambda}(t)) \rightarrow \infty$  only if  $f_i(\hat{x}(t)) \rightarrow 0^-$  for one or more  $i \in \{1, \dots, m\}$ , Alexandrov's theorem [9] implies that

$$\lim_{f_i(\hat{x}(t)) \rightarrow 0^- \text{ for one or more } i} \frac{d}{dt} \mathcal{B}(\hat{x}(t), \hat{\lambda}(t)) < 0, \quad (6)$$

which asserts that  $\mathcal{B}(\hat{x}(t), \hat{\lambda}(t))$  must decrease along the system trajectories when these trajectories are near the boundary. Thus,  $f_i(\hat{x}(t)) < 0$ ,  $i = 1, \dots, m$  for all  $t \geq 0$ . The significance of this conclusion is that the evolution of (4) can be stopped at any time instant with a guaranteed feasible solution.

**Implementation.** Although system (4) is continuous-time, the above-mentioned properties (i.e., convergence and constraint-handling) are approximately maintained when system (4) is implemented in discrete time by making use of the difference quotient and with a sufficiently small sampling period. Thus, to solve problem (2), one can run system (4) until the available computation time is exhausted (that may not be known in advance), and the solution is sub-optimal and guaranteed to enforce the constraints whenever the evolution of system (4) is terminated. This allows the designer to implement optimization-based control tasks with a small sampling period (and consequently with a minimum degradation in performance), while maintaining optimality and constraint-handling capabilities. It is noteworthy that warm-starting can improve convergence of system (4).

### III. EXPERIMENTAL RESULTS

The objective of this section is to validate the proposed optimization approach and assess its effectiveness. The experiments are carried out on an Intel(R) Core(TM) i7-7500U CPU 2.70 GHz with 16.00 GB of RAM. We use YALMIP toolbox to implement the optimization computations.

We consider a case study where two control tasks are executed on a single processing unit. Tasks  $\tau_1$  and  $\tau_2$  implement MPC to control DC motors #1 and #2 given in [10], respectively, such that the control inputs belong to the interval  $[-10, 10]$ . The desired sampling period for both tasks is 20 ms. However, we observe (from 2000 runs) that  $\ell_1 = \ell_2 \approx 150$  ms, implying that the schedulability condition (1) cannot be satisfied with the desired sampling periods. To satisfy (1) we would need, for instance, that the tasks execute every 300 ms.

We consider the following cases: i) MPC with sampling period  $\Delta T = 20$  [ms], which is desired but unimplementable; ii) MPC with sampling period  $\Delta T = 300$  [ms] to satisfy condition (1); and iii) MPC with sampling period  $\Delta T = 20$  [ms] and the proposed optimization approach implemented with  $\sigma = 10$  and  $\beta = 10^5$ . For comparison purposes, we use the Integral Square Error (ISE) index which is  $\text{ISE}(t) \triangleq \int_0^t e(\eta)^2 d\eta$ , where  $e(t)$  is the tracking error and the integration is performed since the start of the experiment at time 0 till the current time  $t$ .

The obtained ISEs for the above-mentioned cases are shown in Fig. 1. As seen in this figure, using a large sampling period degrades the performance by 39% for task  $\tau_1$  and by 62% for task  $\tau_2$ . However, MPC with the proposed optimization approach yields better performance by computing a sub-optimal but feasible solution every 20 ms. More precisely, performance degradation with the proposed optimization approach is 4% for task  $\tau_1$  and 9% for task  $\tau_2$ .

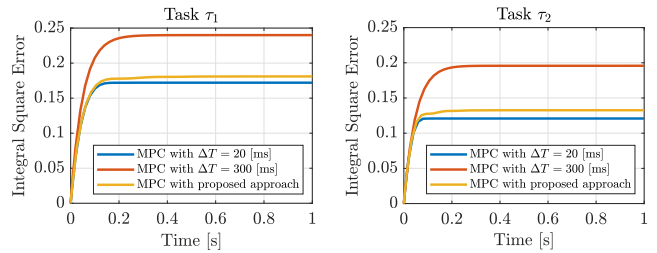


Fig. 1: Obtained ISEs for considered cases. MPC with the proposed optimization approach has 32% gain in the obtained ISE for task  $\tau_1$  and 48% gain in the obtained ISE for task  $\tau_2$  over the MPC with sampling period  $\Delta T = 300$  [ms].

### IV. CONCLUSION AND FUTURE WORK

Limited capacity to perform computations is a common constraint in today's CPSs. We proposed an approach to implement optimization-based control tasks despite the variability and unpredictability of available computing time. Experiments were carried out to assess the effectiveness of the proposed approach in satisfying performance requirements and real-time schedulability conditions. Future work will consider how the proposed optimization approach can empower optimization-based control schemes to deal with time-varying constraints.

### REFERENCES

- [1] G. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*. Springer US, 2011.
- [2] P. Pazzaglia, A. Hamann, D. Ziegenbein, and M. Maggio, "Adaptive design of real-time control systems subject to sporadic overruns," in *Proc. 2021 Design, Automation and Test in Europe Conference and Exhibition*, Grenoble, France, Feb. 1–5, 2021, pp. 1887–1892.
- [3] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear model predictive control: Towards new challenging applications*, L. Magni, D. M. Raimondo, and F. Allgower, Eds. Springer Berlin Heidelberg, 2009, pp. 345–369.
- [4] B. Wang, J. Huang, C. Wen, J. Rodriguez, C. Garcia, H. B. Gooi, and Z. Zeng, "Event-triggered model predictive control for power converters," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 1, pp. 715–720, Jan. 2021.
- [5] D. Liao-McPherson, M. M. Nicotra, and I. Kolmanovsky, "Time-distributed optimization for real-time model predictive control: Stability, robustness, and constraint satisfaction," *Automatica*, vol. 117, Jul. 2020.
- [6] M. M. Nicotra, D. Liao-McPherson, and I. V. Kolmanovsky, "Embedding constrained model predictive control in a continuous-time dynamic feedback," *IEEE Transactions on Automatic Control*, vol. 64, no. 5, pp. 1932–1946, May 2019.
- [7] M. Hosseinzadeh, B. Sinopoli, I. Kolmanovsky, and S. Baruah, "ROTEC: Robust to early termination command governor for systems with limited computing capacity," *Systems & Control Letters*, vol. 161, Mar. 2022.
- [8] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill Publishing, LLC, 2017.
- [9] A. D. Alexandrov, *Convex Polyhedra*. Springer-Verlag Berlin Heidelberg, 2005.
- [10] D. Roy, C. Hobbs, J. H. Anderson, M. Caccamo, and S. Chakraborty, "Timing debugging for cyber-physical systems," in *Proc. 2021 Design, Automation and Test in Europe Conference and Exhibition*, Grenoble, France, Feb. 1–5, 2021, pp. 1893–1898.