Optimal Sensor and Actuator Scheduling in Sampled-Data Control of Spatially Distributed Processes

Da Xue and Nael H. El-Farra

*University of California, Davis, CA 95616 USA (e-mail: nhelfarra@ucdavis.edu)

Abstract: This work presents an optimization-based methodology for the placement and scheduling of measurement sensors and control actuators in spatially-distributed processes with low-order dynamics and discretely-sampled output measurements. Initially, a sampled-data observer-based controller, with an inter-sample model predictor, is designed based on an approximate finite-dimensional system that captures the infinite-dimensional system’s dominant dynamics. An explicit characterization of the interdependence between the stabilizing locations of the sensors and actuators and the maximum allowable sampling period is obtained. Based on this characterization, a constrained finite-horizon optimization problem is formulated to obtain the sensor and actuator locations, together the corresponding sampling period, that optimally balance the tradeoff between the control performance requirements on the one hand, and the demand for reduced sampling, on the other. The objective function penalizes both the control performance cost, expressed in terms of the response speed and the control effort, and the sampling cost, expressed in terms of the sampling frequency. The optimization problem is solved in a receding horizon fashion, leading to a dynamic policy that varies the sensor and actuator spatial placement, together with the sampling period, over time. The developed methodology is illustrated through an application to a simulated diffusion-reaction process example.

Keywords: Sampled-data control, sensor and actuator placement, receding horizon optimization, scheduling, spatially-distributed systems.

1. INTRODUCTION

Despite the extensive body of research work on control of spatially-distributed systems, the problem of designing sampled-data feedback controllers for spatially-distributed processes has received relatively limited attention (e.g., see Logemann et al. (2005); Cheng et al. (2009); Fridman and Blighovsky (2010) for some notable exceptions). This is an important problem given the prevalent use of digital sensors and controllers in industrial control systems. To date, the overwhelming majority of studies on the analysis and design of sampled-data control systems have focused primarily on lumped parameter systems modeled by ordinary differential equations (e.g., see Chen and Francis (1995); Nesic and Grune (2005); Hu et al. (2007); Naghshtabrizi et al. (2008); Fujioka (2009); Hu and El-Farra (2011)).

An effort to address the sampled-data control problem for spatially-distributed systems was undertaken in a series of earlier works (e.g., Yao and El-Farra (2011, 2012)). The main idea was to include an approximate finite-dimensional model of the infinite-dimensional system in the controller to provide estimates of the dominant slow states that compensate for the unavailability of output measurements between sampling times, and to update the model states using the actual measurements at each sampling time. A key objective of this model-based control approach was to guarantee closed-loop stability with minimal sampling frequency. This is an appealing goal in situations where technological constraints on the measurement sensing techniques make it difficult or costly to collect measurements frequently. To this end, an exact characterization of the maximum allowable sampling period required for stabilization was obtained in the earlier studies and found to depend on the spatial placement of the measurement sensors and control actuators. An implication of this finding is that one could use this characterization to identify the set of feasible sensor and actuator locations that can achieve stabilization with minimal sampling frequencies.

While this is important from a sampling cost savings standpoint, the resulting actuator and sensor placement does not consider the impact of discrete measurement sampling on closed-loop performance. It is well known, for example, that faster sampling rates are typically required to enhance the control system performance. Ultimately, there is a need to resolve the inherent conflict that arises between the tight restrictions on sampling rates which are required to maintain the control system performance on the one hand, and the usual demand for limited sampling frequency which is desired to minimize measurement costs.

Motivated by these considerations, we present in this work an optimization-based approach for the integration of control and actuator/sensor scheduling in spatially-distributed processes with discretely-sampled output measurements. The objective is to simultaneously optimize...
the control system performance and the sampling costs by dynamically scheduling the sensor and actuator locations, as well as the sampling period, subject to the appropriate stability constraints. The rest of the paper is organized as follows. Following some preliminaries in Section 2, an observer-based controller that relies on an inter-sample model predictor and a fixed sensor-actuator placement is introduced in Section 3, along with a parametric characterization of its closed-loop stability region. This characterization is incorporated as a constraint within a finite-horizon optimization problem that is formulated in Section 4, where the cost function includes explicit penalties on the response speed, the control action and the sampling frequency. A receding horizon strategy for implementing the optimization problem solution is then devised in Section 5 to determine the optimal placement and scheduling of the sensors and actuators and the optimal sampling rate. Finally, a simulation case study is presented in Section 6.

2. PRELIMINARIES

We consider spatially-distributed processes modeled by highly-dissipative infinite-dimensional systems. An example of this class of systems, which is introduced in this section to illustrate the subsequent theoretical development, are systems described by parabolic PDEs:

$$\frac{\partial x}{\partial t} = \alpha \frac{\partial^2 x}{\partial z^2} + \beta x + \omega \sum_{i=1}^n b_i(z) u_i$$

(1)

subject to the boundary and initial conditions:

$$\begin{align*}
    x(z, t) &= \bar{x}(z, t) = 0, \quad x(z, 0) = \bar{x}_0(z) \\
    y_i(t) &= \int_0^\pi q_i(z) x(z, t) dz, \quad i \in \{1, 2, \ldots, l\}
\end{align*}$$

(2)

where $x(z, t) \in \mathbb{R}$ is the process state variable, $z \in [0, \pi]$ is the spatial coordinate, $t \in [0, \infty)$ is the time, $u_i \in \mathbb{R}$ is the $i$-th manipulated input, $n$ is the number of manipulated inputs, $b_i(\cdot)$ is the actuator distribution function, $y_i$ is the $i$-th measured output, $q_i(\cdot)$ is the $i$-th sensor distribution function, and $\bar{x}_0(z)$ is a smooth function of $z$. The PDE of (1), subject to the boundary and initial conditions of (2) can be formulated as an infinite-dimensional system with the following state-space representation:

$$\begin{align*}
    \dot{x}(t) &= A x(t) + B u(t), \quad x(0) = x_0 \\
    y(t) &= Q x(t)
\end{align*}$$

(3)

where $x(t) = x(z, t)$, $t > 0$, $z \in [0, \pi]$, is the state function defined on an appropriate Hilbert space, $H = L_2(0, \pi)$, endowed with the proper inner product and norm; $A$ is the differential operator; $B$ and $Q$ are the input and output operators, respectively; $u = [u_1, \ldots, u_n]^T$, $y = [y_1, \ldots, y_l]^T$, and $x_0 = \bar{x}_0(z)$. Applying model decomposition techniques, the infinite-dimensional system can be decomposed as follows:

$$\begin{align*}
    \dot{x}_s &= A_s x_s + B_s u, \quad x_s(0) = P_s x_0 \\
    \dot{x}_f &= A_f x_f + B_f u, \quad x_f(0) = P_f x_0 \\
    y &= Q_s x_s + Q_f x_f
\end{align*}$$

(4)

where $x_s = P_s x$ is the state of a finite-dimensional system that describes the evolution of the slow (possibly unstable) eigenmodes; $x_f = P_f x$ is the state of an infinite-dimensional system that captures the evolution of the fast (stable) eigenmodes; and $P_s$ and $P_f$ are the orthogonal projection operators, where $A_s = P_s A$, $B_s = P_s B$, $A_f = P_f A$, and $B_f = P_f B$. To simplify the presentation of the main results, we assume that the number of actuators and sensors is the same and equal to the number of the dominant modes (i.e., $n = l = m$).

Based on the above decomposition, the following approximate finite-dimensional system which describes the temporal evolution of the slow (possibly unstable) eigenmodes can be derived using Galerkin’s method:

$$\begin{align*}
    \dot{\hat{a}}_s &= A_s \hat{a}_s + B_s(z_s) u, \quad \dot{\hat{y}}_s = Q_s(z_s) \hat{a}_s \\
    \dot{\hat{a}}_s &= A_s \hat{a}_s + B_s(z_s) u, \quad \dot{\hat{y}}_s = Q_s(z_s) \hat{a}_s
\end{align*}$$

(7)

where $a_s = [a_1 \cdot \cdot \cdot a_m]^T \in \mathbb{R}^m$; $a_i$ is the amplitude of the $i$-th eigenmode; $\dot{\hat{y}}_s$ is the output; $A_s$ is an $m \times m$ diagonal matrix of the form $A_s = \text{diag}\{\lambda_j\}$ containing the slow eigenvalues in the spectrum of $A$; $B_s$ is an $m \times m$ input matrix whose individual elements are parameterized by the locations of the control actuators $z_s$; $\hat{Q}_s$ is an invertible $m \times m$ output matrix whose individual elements are parameterized by the locations of the measurement sensors $z_s$ (the invertibility assumption can be ensured by proper selection of the sensors’ locations). The reduced-order system of (7) will be used in the next section to design a model-based output feedback controller that achieves stabilization using discretely-sampled output measurements.

3. OBSERVER-BASED CONTROL USING AN INTER-SAMPLE MODEL PREDICTOR

We consider an output-feedback control structure in which the output measurements collected by the sensors are available to the controller only at discrete sampling times. A dynamic state observer is used to generate an estimate of the state which is used to compute the control action. Owing to the unavailability of output measurements between sampling times, an inter-sample model predictor is used to generate an estimate of the output of the system between sampling times. This estimate is used by the observer and is re-set using the actual output when it becomes available at the next sampling time. The implementation of this combined control and update strategy is given by:

$$\begin{align*}
    u(t) &= K \eta(t), \quad t \in [t_k, t_{k+1}) \\
    \dot{\hat{y}}(t) &= \hat{A} \hat{y}(t) + \hat{B}_s u(t) + \hat{L}(\hat{y}_s(t) - \hat{Q} \eta(t)) \\
    \dot{\hat{a}}_s(t) &= \hat{A} \hat{a}_s(t) + \hat{B}_s(z_s) u(t) \\
    \hat{y}_s(t) &= \hat{Q}_s(z_s) \hat{a}_s(t) \\
    \hat{a}_s(t_k) &= \hat{Q}_s^{-1} \eta(t_k), \quad k \in \{1, 2, \ldots\} \\
    t_{k+1} &= t_k + \Delta
\end{align*}$$

(8)

where $\eta$ is the state of the observer; $\hat{A}_s, \hat{B}_s, \hat{Q}_s$ are approximate models of $A_s, B_s$ and $\hat{Q}_s$, respectively; $K$ and $\hat{L}$ are the feedback controller and observer gains, respectively; $\hat{a}_s$ and $\hat{y}_s$ are the state and output of the inter-sample model predictor; $t_k$ denotes the $k$-th sampling time, and $\Delta > 0$ is the sampling period. For simplicity, we assume that $\hat{Q}_s = Q_s$ (i.e., no uncertainty in $z_s$).

The following theorem provides an exact characterization of the closed-loop stability properties under the observer-based controller of (8) with a periodic sampling strategy. The proof is conceptually similar to the one in Garcia et al. (2014) and is omitted for brevity.

**Theorem 1.** Referring to the closed-loop system of (7)-(8), the origin is exponentially stable if and only if:

$$\lambda_{\text{max}}(M(K, L, \Delta, \hat{A}_s, \hat{B}_s(z_s), A_s, B_s, \hat{Q}_s(z_s))) < 1,$$

(9)

where $\lambda_{\text{max}}$ is the largest eigenvalue magnitude of the matrix $M$ which is given by:

$$M = \begin{bmatrix}
    0 & \hat{A}_s & \hat{B}_s(z_s) \\
    A_s & \hat{A}_s & \hat{B}_s(z_s) \\
    0 & \hat{A}_s & \hat{B}_s(z_s) \\
\end{bmatrix}$$

and $\hat{A}_s = \hat{Q}_s^{-1} \hat{K} \hat{Q}_s^{-1}$, $\hat{B}_s(z_s) = \hat{Q}_s^{-1} \hat{B}_s(z_s)$, $\hat{Q}_s(z_s) = \hat{Q}_s(z_s)$. The proof is similar to the one in Garcia et al. (2014) and is omitted for brevity.
where

\[ M = I \sigma \Delta \Lambda , \quad (10) \]

where

\[ I_s = \begin{bmatrix} \tilde{A} & \tilde{B} & \tilde{K} \\ LQ \end{bmatrix} \]

(11)

where \( I_{m \times m} \) is the \( m \times m \) identity matrix, \( O_{m \times m} \) is the \( m \times m \) zero matrix, and \( \Lambda \) is the closed-loop matrix given by:

\[ \Lambda = \begin{bmatrix} A_s \\ LQ \end{bmatrix} \]

Remark 1. The stability condition of (9) provides an explicit characterization of the interdependence between the process and model parameters, the controller and observer gains, the sampling period, and the sensor and actuator locations in influencing closed-loop stability. For example, for fixed model and controller parameters, this characterization can be used to determine how the maximum allowable sampling period depends on the spatial placement of the sensors and actuators. This characterization is important as it provides the basis for the optimization formulation introduced in the next section.

4. AN OPTIMIZATION-BASED FORMULATION FOR SENSOR AND ACTUATOR PLACEMENT

While the stability condition in Theorem 1 provides a useful tool that can be used to analyze how different sensor and actuator locations influence the size of the maximum allowable sampling period, it leaves open the question of how to best place the sensors and actuators in a way that optimally balances the desire for reduced sampling frequency with the demand for improved control system performance. Furthermore, the implementation of the observer-based controller of (8) involves the use of a constant sampling rate which is not always the best choice, especially in the presence of unexpected external disturbances and changing operating conditions. In this section and the next, we describe an optimization-based formulation that aims to address both problems.

We consider the following finite-horizon optimization problem in which the sampling period, the sensor and actuator locations at any given time, and the optimization problem. The solution to the optimization problem is the optimal triple \((z^k_s, \Delta k, \Delta k)\) that minimizes the total cost \( J \) over the given horizon \( H \).

Remark 2. While the optimal actuator/sensor placement problem for spatially-distributed systems has been the subject of significant prior work (e.g., see Antoniades and Christofides (2000); Demetriou and Kazantzis (2004); Armaou and Demetriou (2006); Iftime and Demetriou (2009)), the emphasis of previous approaches has been either on meeting certain controllability and observability requirements, or on optimizing the performance of the closed-loop system (e.g., in terms of the response speed and control effort) without consideration of the costs or associated with discrete measurement sampling. The formulation proposed in (13) represents a departure from conventional approaches by aiming to balance control performance considerations with the demand for reduced sampling costs associated with discrete measurement sampling. Specifically, the objective function in (13) consists of a performance cost, \( J_p \), which imposes penalties on the model output and the control action, and a sampling cost, \( J_s \), which penalizes frequent sampling. A simple choice for the sampling cost is to set \( \beta = w_\Delta / \Delta \), which ensures that faster sampling rates incur larger penalties. Note that \( J_p \) is expressed in terms of the model output since the actual output is unavailable between sampling times.

Remark 3. The choice of the various penalty weights in the optimization formulation should be made to reflect the relative significance of the different costs. For example, imposing larger penalties on the sampling frequency suggests an increased emphasis on the sampling cost relative to the control performance. On the other hand, for systems where sampling costs are negligible, more emphasis is placed on the controller performance.

5. DYNAMIC ACTUATOR/SENSOR SCHEDULING USING RECEIVING HORIZON OPTIMIZATION

Referring to the implementation of the optimization formulation in (13), one possible strategy is to solve the problem once at the beginning of the horizon, apply the resulting optimal sensor and actuator locations and sampling period over the entire horizon length, and then repeat the optimization when the end of the horizon is reached. While this strategy reduces the computational load associated with repeated off-line optimization (especially if \( H \) is large), it does not take into account the fact that the cost function is dependent on the model state trajectory, and the fact that the model state is to be updated at each
sampling time. It may also limit the ability of the process
to respond in a timely fashion to changes in operating
conditions by limiting the frequency of feedback from the
process. These considerations call for a receding horizon
implementation strategy, where (13) is solved on-line at
each tk, and the resulting sensor and actuator locations, z^k_s
and z^k_a, together with the corresponding stabilizing
sampling period, ∆_k, are implemented only for a single
sampling period within the horizon. By the end of the
sampling interval at t = tk + ∆_k, a new sampled output
measurement is sent from the sensor to the controller and
used to update the model state as follows:

\[
(z_s(t), z_a(t), ∆(t)) = (z^{k+1}_s, z^{k+1}_a, ∆^{k+1}_s)
\]

With the updated model state, the optimization problem
of (13) is re-solved to determine (z^{k+1}_s, z^{k+1}_a, ∆^{k+1}_s) at
\(t = tk+1 = tk + ∆_k\). The newly obtained solution is then
implemented for \(t \in [tk+1, tk+1 + ∆_k+1]\) and the problem is
re-solved at end of the sampling period. Algorithm 1
summarizes the on-line receding horizon strategy.

**Remark 4.** The repeated on-line optimization approach
described in Algorithm 1 leads to a time-varying schedule
for moving the sensor and actuator locations as well as
the sampling rate. Compared with the observer-based
controller in (8) with a constant sampling rate, the
time-varying scheduled sensor and actuator locations and sam-
pling rates obtained through the optimization formulation
allow the process to adaptively respond to expected
changes in operating conditions.

**Remark 5.** The receding horizon implementation of the
optimization formulation bears resemblance to the imple-
mentation of MPC policies, where a finite-horizon opti-
mization problem is re-solved at every sampling time, and
the optimal solution is implemented for the duration of the
first sampling period only, at the end of which new output
measurements are obtained and fed back to the
controller to update the model state and correct possible
deviations in the model state trajectory. This feedback-
based approach is appealing since it provides some robust-
ness to process-model mismatch and changes in operating
conditions due to disturbances. Note, however, that unlike
conventional MPC schemes, the frequency at which the
optimization problem is re-solved is not fixed. It is rather
determined online as part of the optimization problem
solution. Furthermore, closed-loop stability is guaranteed
through the constraint in (9) and is therefore independent of
the horizon length.

**Remark 6.** From an implementation standpoint, the hori-
zon length should be chosen such that H > ∆. A suitable
choice for H can be made on the basis of the characteriza-
tion of the feasible region in terms of z_s, z_a and ∆ obtained
prior to online implementation. Specifically, by analyzing the λ_max(M) over a specified range of sensor and actuator
locations, an upper bound on the feasible sampling period
can be obtained. In other words, the maximum allowable
sampling period, ∆_max, can be viewed as a function of z_s
and z_a, where the stability condition that λ_max(M) < 1
is satisfied. Then by considering a range of possible sensor
and actuator locations, a corresponding range of possible
sampling periods could be estimated and used to choose

1. characterize ∆_max(z_s, z_a) based on the stability condi-
tion of (9);
2. specify the penalty weights W_y, W_u and w_Δ;
3. initialize ̂y_s(t_0) = ̂y_s(t_0), ̂a_s(t_0) = η(t_0) = Q_1^-1 ̂y_s(t_0);
4. solve (13) to determine z^1_s, z^1_a and ∆_0;
5. place z_s = z^1_s and z_a = z^1_a;
6. start system operation;
7. sample output measurements ̂y_s(t) at t = Δ_t;
8. set k = 1;
9. solve equation (13) to determine z^k_s, z^k_a and ∆_k;
10. relocate z_s = z^k_s and z_a = z^k_a;
11. while t_k ≤ t < t_k + ∆_k do
   | z_s(t) = z^k_s and z_a(t) = z^k_a;
   | u(t) = K(η(t));
end
12. if t = t_k + ∆_k then
   | sample output measurements ̂y_s(t);
   | update model state ̂a_s(t) = Q_1^-1 ̂y_s(t);
   | k = k + 1;
   | goto step 9
end

Algorithm 1. A receding horizon implementation strategy for
actuator/sensor scheduling.

A suitable horizon length, where ∆_max serves as a lower bound on H (see Section 6 for an analysis of the impact
of the horizon length).

### 6. SIMULATION EXAMPLE

We consider a non-isothermal diffusion-reaction process
described by the following parabolic PDE:

\[
\frac{∂\bar{x}}{∂t} = \frac{∂^2 \bar{x}}{∂z^2} + ([β_T + θ_1]γe^{-γ} - β_U)\bar{x} + β_Ub(z)u(t)
\]

subject to Dirichlet boundary conditions as in (2), where \(\bar{x}\)
is a dimensionless process temperature, the manipulating
input u is a dimensionless temperature of the cooling
medium, \(β_T = 80.0, γ = 2.0, β_U = 1.66\) are nominal
process parameters, \(θ_1 = 0.01\) is a parametric uncertainty
in the heat of reaction, and \(b(z)\) is the actuator distribution
function. It can be verified that the operating steady-
state z(x, t) = 0 (with u = 0) is unstable. The control
objective is to stabilize the temperature profile at this unstable,
spatially uniform steady-state by manipulating the
temperature of the cooling medium with minimal output
measurement sampling. A point control actuator and a
single point measurement sensor (with finite support) are
assumed to be available. We consider the first eigenvalue of
the differential operator to be dominant and use Galerkin’s
method to derive an uncertain ODE model which is used
for the synthesis of the controller. It was verified that the
controller under continuous sampling stabilizes the
closed-loop system. In the following simulation studies, the
controller and observer gains are chosen as K = 15 and
L = 50, respectively.

#### 6.1 Characterizing the closed-loop stability region

Following the proposed methodology, the first step is to
characterize the stability region in terms of the sensor and
actuator locations and the sampling period. We begin by analyzing the effect of the sensor location on closed-loop stability, the sampling and performance costs when the actuator is placed at the middle of the spatial domain at $z_a = \pi/2$.

Fig.1(a) shows a contour plot of $\lambda_{\text{max}}(M(z_s, \Delta))$ where the region enclosed by the unit-contour line (the uncolored region) is the stable region, while the colored region represents the zone where $\lambda_{\text{max}}(M(z_s, \Delta)) > 1$ and therefore is the unstable region. The implication of this plot is that for a specific sensor location, there is a maximum allowable sampling period beyond which instability would occur. Fig.1(b) traces the unit-contour line in Fig.1(a) which defines the maximum allowable sampling period, $\Delta_{\text{max}}(z_s)$, as a function of $z_s$. The shape of $\Delta_{\text{max}}(z_s)$ is symmetric and suggests that prolonging the maximum allowable sampling period requires placing the sensor closer to the boundaries. Fig.1(c) shows a snapshot of the breakdown of the total cost (defined in (13)) at the first sampling time, $t_1 = 0.208$, when the optimization problem is solved on-line for the first time in the operating stage. The weighting parameters are chosen as $W_y = W_u = 150$, $w_\Delta = 1$, and $H = 2$ (which is approximately twice the maximum allowable sampling period achievable in the actuator spatial domain). The red line shows the sampling cost, $J_s$, as a function of $z_s$, while the blue curve depicts the control performance cost, $J_p$, as a function of $z_s$. As expected, the dependence of $J_s$ on $z_s$ is opposite to that of $\Delta_{\text{max}}(z_s)$ shown in Fig.1(b) since longer sampling periods translate into lower sampling cost. It can be seen that both the sampling and performance costs exhibit similar qualitative dependence on $z_s$, where the cost is largest at the center, $z_s = \pi/2$ and decreases as the sensor is moved closer to the boundary (even though the performance cost appears to depend only weakly on $z_s$ as can be seen in Fig.1(d)). This suggests that no tradeoff between the two costs exists, and that reducing the total cost favors locating the sensor close to the boundary. This leads to a static (i.e., fixed) sensor placement instead of a dynamic scheduling policy.

Similar to the foregoing analysis, the feasible region with respect to the actuator location can be characterized. Fig.2(a) shows a contour plot of $\lambda_{\text{max}}(M(z_a, \Delta))$ for a fixed sensor location, whereas Fig.2(b) plots $\Delta_{\text{max}}(z_a)$ for different sensor locations. The analysis here shows the same trend observed for the sensor placement problem regarding the feasible region, i.e., moving the actuator closer to the boundary helps increase the maximum allowable sampling period (i.e., reduce the sampling cost). Moreover, it can be seen that moving the sensor away from the center enlarges the feasible region leading to longer permissible sampling periods.

Fig.2. Plot (a): Contour plot of $\lambda_{\text{max}}(M(z_a, \Delta, z_s = \pi/2))$. Plot (b): Maximum allowable sampling period as a function of actuator locations, $\Delta_{\text{max}}(z_a)$, for various sensor locations.

6.2 Optimization-based actuator scheduling strategy

The stability region characterization in Fig.2 is used in this section to formulate and solve the optimization problem of (13). The weighting parameters and the optimization horizon length are kept the same as in Section 6.1. Note that when the maximum allowable sampling period is used, the closed-loop system is only critically stable. Therefore, to enforce asymptotic stability, the actual sampling period needs to be reduced slightly in practice. In this section, the sampling period used is $0.7\Delta_{\text{max}}$.

Fig.3 shows the influence of the actuator location on the total cost and its breakdown at $t = 0.208$ and at $t = 4$, respectively. It can be seen that, in both cases, minimizing the sampling cost favors moving the actuator closer to the boundary whereas minimizing the performance cost favors moving it away from the boundary, leading to an optimal location in between that minimizes the total cost. Comparing the two plots, it can also be seen that the sampling cost does not change over time (note the different scales), whereas the performance cost decreases as time goes on. This is expected since the sampling cost does not depend on the model state, and therefore is unaffected by the sampling and model updates. But since the controller is designed to be stabilizing, the model output continues to decay over time, leading to smaller performance costs. Moreover, at early times when the model output is far away from its steady state, the performance cost is more dominant, but as the output converges, the sampling cost picks up and becomes more dominant at later times. As a result, the shape of the total cost varies over time, causing the optimal actuator location to vary with time.

Fig.4 shows the results when the optimization problem in (13) is solved in a receding horizon fashion. Fig.4(a) depicts the closed-loop state profile when $z_a = \pi/10$, while Figs.4(b)-(d) compare the closed-loop state trajectory, the optimal actuator schedules, and the sampling frequencies.
for various sensor locations. Fig.4(b) shows that the sensor location slightly affects the closed-loop state trajectory. However, it does influence the optimal actuator schedule as can be seen in Fig.4(c); and moving the sensor away from the center helps reduce the sampling frequency as can be seen in Fig.4(d). Regarding the optimal actuator schedules, it can be seen from Fig.4(c) that all schedules start at the middle location which is the one that optimizes performance cost. Moving the sensor away from the center helps reduce the sampling frequency as can be seen in Fig.4(d); and moving the sensor away from the center helps reduce the sampling frequency as can be seen in Fig.4(d).

Table 1. Cost comparisons for different sensor locations.

<table>
<thead>
<tr>
<th>$z_s$</th>
<th>performance cost</th>
<th>sampling cost</th>
<th>total cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\pi/2$</td>
<td>2410.2</td>
<td>20.63</td>
<td>2430.8</td>
</tr>
<tr>
<td>$\pi/5$</td>
<td>2165.2</td>
<td>16.10</td>
<td>2181.3</td>
</tr>
<tr>
<td>$\pi/10$</td>
<td>1865.5</td>
<td>13.60</td>
<td>1879.1</td>
</tr>
</tbody>
</table>

Fig. 3. Snapshots of the sampling cost, the performance cost and the total cost as functions of $z_s$, at $t = 0.208$ (plot (a)) and at $t = 4$ (plot (b)).

Fig. 4. Plot (a): Closed-loop state profile with $z_s = \pi/10$. Plot (b): Comparison of closed-loop state trajectory for various sensor locations. Plot (c): Comparison of the actuator scheduling policies for different sensor locations. Plot (d): Comparison of the sampling times for different sensor locations.

REFERENCES


