A Near-Optimal Sensor Scheduling Strategy for an ON–OFF Controller With an Expensive Sensor

Biju Edamana and Kenn R. Oldham, *Member, IEEE*

***Abstract*—This paper describes an efficient method for scheduling an energy-consuming sensor sparingly in combination with an ON–OFF controller, specifically for a finite horizon control problem in which only end states are critical. In certain low-power applications, such as autonomous microrobotics, ON–OFF controllers can be very efficient in operating piezoelectric actuators (and other capacitive actuation schemes) compared to traditional analog and pulsewidth modulation controllers. However, with existing sensing circuitry, sensing at the same frequency as control can be prohibitively expensive, because energy consumption in the sensing circuitry may be comparable or even much higher than energy consumption for actuation. Instead, a method is presented for best scheduling a limited number of sensor measurements and updates to control inputs during a finite horizon ON–OFF control problem, in response to Gaussian disturbances and measurement noise. To simplify the problem, a lower bound for the expected value of a quadratic error function of the end states is found, which permits rapid evaluation of candidate sensor times. When actuator energy consumption is incorporated in the optimization, this produces a numerically efficient near-optimal strategy for determining best measurement times and updates to the control input sequence.**

***Index Terms*—Integer programming, Kalman filters, microactuators, microelectromechanical devices, ON–OFF control, piezoelectric devices, switched systems.**

I. Introduction

THIS paper describes a method for selecting intermittent measurement along with input update times and values for ON–OFF control of linear systems, so as to minimize state error under sensing and control constraints imposed by strict energy limits. This extends previously developed open-loop ON–OFF controllers for low-power capacitive microactuators by including feedback at a limited number of sample times. The objective of previously developed controllers was to minimize power consumption of capacitive (piezoelectric or electrostatic) actuators to reach specified final system states. The chief application for optimizing finite duration ON–OFF control inputs was evaluation of autonomous walking microrobot feasibility, with ON–OFF inputs intended to produce a quasi-static walking gait by controlling rotational motions of microrobotic leg joints driven by piezoelectric actuators. During each step, actuators may be rotated to prescribed angles in a prescribed time using minimum energy by solving an integer programming problem; the resulting ON–OFF inputs minimize the number of times the actuator is turned ON while staying within a specified bound on final position error [1]. This previous work done by the authors is an alternative to existing control strategies for piezoelectric actuators proposed earlier by Main *et al.* [2] and Campolo *et al.* [3]. In the previous work of authors, sensors were avoided due to the high energy consumption of existing microelectromechanical systems position sensors, such as piezoresistive or capacitive deflection sensors. As a next step in low-power controller development, in this paper an optimal feedback strategy is introduced that uses a sensor sparingly to improve on error performance within a strict sensor energy budget.

Since microactuator motions are often subject to disturbances, some level of feedback is generally desirable if specified displacements are to be reliably achieved, but limiting the frequency of feedback can still be useful for keeping system power consumption low. As a frame of reference, capacitances of the proposed thin-film lead-zirconate-titanate microactuators for microrobotics are often on the order of 0.1–1 nF [4]. In [1], the authors proposed use of optimal ON–OFF control strategies requiring nominally 40–400 nJ per charge/discharge cycle at 20 V. In comparison, state-of-the-art commercial low-power capacitive sensing circuits with A/D conversion require the equivalent of about 500 nJ per sample acquisition (50 μW for 100 samples/s [5]), and piezoresistors at this scale would need to be operated at less than 2% duty cycle to achieve comparable energy consumption. Similar sensing circuit power consumption was reported in [6] and [7]. Experimental circuits based on semi-analog, duty-cycle-output processing appear capable of dramatically reducing energy consumption [8], and sensors based on direct voltage sensing from piezoelectric elements have likewise been demonstrated with much smaller energy consumption per sample [9]. However, energy consumption per sample taken by these techniques still ranges from about 2% to 85% of that required for each piezoelectric actuator used, and these latter techniques are less directly compatible with low-power microcontrollers and displacement sensing, respectively. Meanwhile, adjusting future ON–OFF control inputs at only a limited number of control input update times, as will be done in this paper, reduces the amount of computation and/or memory required to perform nonlinear programming or reference a lookup table to optimize control inputs.

Thus, this paper examines how to best perform ON–OFF control of systems when only a small number of feedback measurements and updates to control inputs are permitted over a finite time period. Several previous studies on sensor scheduling have

Manuscript received December 26, 2011; revised May 29, 2012; accepted September 6, 2012. Date of publication November 21, 2012; date of current version January 17, 2014. Recommended by Technical Editor D. Sun.

The authors are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: bij@umich.edu; oldham@umich.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMECH.2012.2226244

some relevance to this study. Sinopoli *et al.* in [10] identified the existence of a critical measurement rate for an unstable system which bounds the error on state estimates. Recently, Mo and Sinopoli in [11] improved the bounds on critical measurement rate. Park and Sahai in [12] proposed the concept of eigenvalue cycle to tackle the same problem. These results are developed in the context of an unreliable, unstable wireless sensor network. Schenato *et al.* [13] developed theories for the linear quadratic Gaussian (LQG) problem, when both control and sensing signals may be lost during transmission. Epstein *et al.* in [14] presented conditions for high probability that the error covariance is bounded above at any point of time in a lossy network. On the other hand, Imer *et al.* minimized a quadratic performance index when the connection between controller and plant is lossy for remote control applications [15]. Note that most of these studies deal with stochastic losses of information, rather than a deterministic evaluation of when to acquire information, though the effect is still to produce infrequently sampled systems.

In [16], Li and Elia discussed a convex optimization strategy that minimizes the maximum error of certain states by having a sensor visit those states frequently. Another group of researchers has been investigating efficient use of multiple sensors by turning ON a smaller set of sensors at a time [17], [18], including for model-predictive applications [19]. This study differs from previous work by restricting controlled system inputs to ON–OFF switching. As the final set of states becomes a function of binary variables, the problem can become a complex one to optimize even with certain simplifying assumptions: first, there is only a single sensor which is turned ON sparingly (a small finite number of times over the duration of motion) and second, the objective function is expressed only in terms of the final states. Following a sensor measurement, adjustment of future ON–OFF inputs to the system is similar to that of certain hybrid or model-predictive controllers. For example, for the controller in this paper, a strategy derived from the receding horizon approach for switching controller optimization of [20] is adopted, while similar model-predictive control algorithms with final costs have been studied in [21]. However, these previous works generally assume fixed sampling rates with frequent measurements, concentrating their studies on the optimal controller design, while in this paper a controller design is combined with the sensor scheduling problem, for the ON–OFF input, single-sensor conditions noted earlier.

This paper is organized as follows. Section II introduces the stochastic LTI system used for this study together with the objective function and constraints on the problem; in Section III, the general optimization problem is reduced to a set of simpler optimizations requiring less computational effort under certain conditions; in Section IV, a three-step strategy is described for finding the best measurement times, input update times, and input sequence for achieving specified performance; Section V discusses four case studies showing the effectiveness of the new solution strategy in various scenarios as well as a plan for implementation of intermediate sensing; Section VI concludes the paper and Section VII discusses the merits and scenarios at which these strategies are beneficial.

II. System Description

The discretized state-space representation of the system to be studied, having n states and a single input and output, is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + B_w w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned} \tag{1}$$

where A, B, B_w, and C are the state matrix, input matrix, disturbance input matrix, and output matrix, respectively, and $w(k) \sim N(0, Q)$ and $v(k) \sim N(0, R)$ represent Gaussian disturbance and measurement noise. The initial state is also assumed to be a normal random variable $x(0) \sim N(0, P_0)$. The system is assumed to be stabilizable and detectable if there are no input and measurement constraints, but for the controller to be implemented the input variable $u(k)$ is assumed to be binary and the transition of input variables happens only at the sampling time instants. Because of this constraint, only discrete points in state space are reachable over finite time, and the objective of the intermittent feedback controller is to minimize the expected value of the quadratic error function from desired final states x_d at time N, as in

$$\min_{u(i)=\{0,1\}} E[(x(N) - x_d)'(x(N) - x_d)] \tag{2}$$

while making measurements sparingly (i.e., taking measurements at $M \ll N$ time steps). Since the energy consuming sensor is turned ON only at a few instances in a given optimization horizon, it is important to choose those times carefully. The question studied here is given a specified number of measurements (or in other words, given an allowed sensor energy consumption), what is the best performance that can be achieved and what are the sampling and update times associated with that performance.

For the case studies illustrating the proposed approach, the model of a prototype microrobot leg joint is used, as shown in Fig. 1(b). The dynamics of this leg can be lumped into a mass-spring-damper system as shown in Fig. 2, where the states of the system are angular displacement and angular velocity of the actuator. Although constant unknown disturbances are expected to be the predominant disturbance in microrobotic applications, the more general problem of randomness entering at each time step is considered. If desired, error due to constant disturbances may be included with appropriate addition of integrators to the system model. Likewise, integration terms related to output can be added if error over the duration of motion is of interest, rather than just the states of the system at the final time.

As a preliminary, consider the deterministic open-loop problem, in which optimal input sequences [values for $u(0)$ to $u(N-1)$] can be obtained by solving an integer programming problem minimizing the quadratic error function $(x(N) - x_d)'(x(N) - x_d)$. Constraints on this problem are the deterministic part of the dynamic constraints given in (1), or the solution when $w(k) = 0$ and $v(k) = 0$. Solving for the nominal open-loop inputs without disturbance is the first step in the solution strategy discussed later. However, to improve the expected value of the quadratic error function in the presence of disturbances

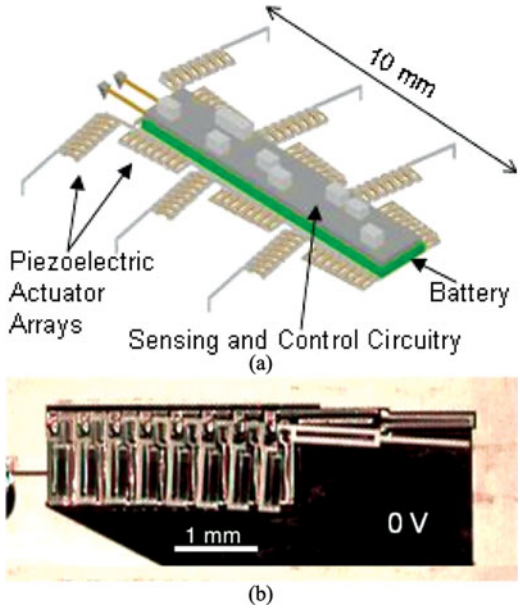

Fig. 1. (a) Concept-drawing of an autonomous microrobot based on thin-film piezoelectric actuator joint arrays. (b) Sample image of a prototype leg joint at 0 V, courtesy U.S. Army Research Laboratory.

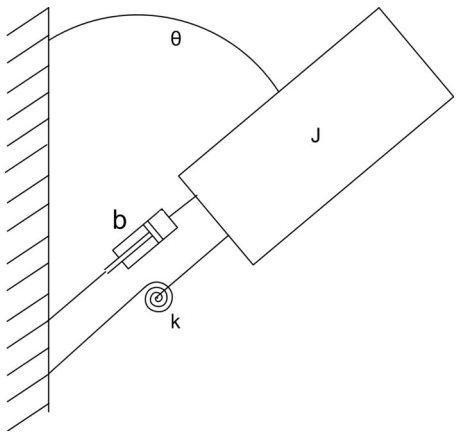

Fig. 2. Dynamic model of a prototype leg joint.

and noise with a fixed, small number of measurements, future ON–OFF inputs may be reoptimized to update the input ON–OFF sequence based on measurement information. Note that updating input sequences more often than the number of measurements does not provide any additional benefit, since no improved knowledge of actual system states is obtained. Also, it is shown later that in certain situations, updating inputs after every measurement has little benefit for system performance and desirable results can be achieved by updating the input sequence fewer times than the number of measurements, thus also avoiding unnecessary online computations or lookup table memory storage.

To introduce the sensing terminology, consider the simplest case where measurements are taken only once over the optimization horizon. Let measurement time be T_l. States of the system based on this measurement can be estimated using an intermittent Kalman filter for any time instant at or after the measurement time. Based on these estimates, the input sequence can be updated at any time $T_u \in [T_l, N]$ by solving a second integer programming for $N - T_u$. The estimates of the states at the update time using Kalman filter equations can be written as

$$\hat{x}(T_u) = A^{(T_u - T_l)} \hat{x}_{T_l/T_l} + \sum_{i=T_l}^{T_u - 1} A^{T_u - 1 - i} B u(i) \tag{3}$$

$$\hat{x}_{T_l/T_l} = \hat{x}(T_l) + P_{T_l} C' [C P_{T_l} C' + R]^{-1} \times (Cx(T_l) - C\hat{x}(T_l) + v(T_l)) \tag{4}$$

$$\hat{x}(T_l) = A^{T_l} \hat{x}(0) + \sum_{i=0}^{T_l - 1} A^{T_l - 1 - i} B u(i) \tag{5}$$

$$P_{T_l} = A^{T_l} P_0 A'^{T_l} + \sum_{i=0}^{T_l - 1} A^i B Q B' A'^i \tag{6}$$

where $\hat{x}_{T_l/T_l}$ is the estimation of states after measurement and $\hat{x}(T_l)$ is the estimation of states before measurement. As mentioned, a new input sequence for the remaining $N - T_u$ time steps is found by solving an integer programming problem that minimizes $(\hat{x}(N) - x_d)'(\hat{x}(N) - x_d)$ where $\hat{x}(N)$ is given as

$$\hat{x}(N) = A^{N - T_u} \hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i} B u(i). \tag{7}$$

III. Problem Reduction

Consider a general problem with a fixed number of measurements. Let the number of measurements be a. Also, let the number of updates be $b \leq a$. The objective function in this case can be written as

$$\min_{T_{l1}, T_{l2} .. T_{la}, T_{u1} .. T_{ub}, u(0) .. u(N-1)} E\{[x(N) - x_d]'[x(N) - x_d]\}. \tag{8}$$

The overall optimization involves finding optimal measurement times, update times, and input sequences that minimize the expected value of the quadratic error function of final states. Note that this objective function does not explicitly minimize combined sensor and actuator energy, as a varying number of samples is difficult to incorporate into the analytical solution. Rather, it should be interpreted as optimizing performance given a fixed level of energy usage (number of measurements) of the sensor during the system response. Comments on comparing sensor energy usage to improvement in performance are included later in Section VII.

If there are no input constraints, this problem can split into two simpler optimization problems involving lesser computational effort. The first optimization problem is to determine the optimal measurement times and the second problem is to evaluate the inputs after the last measurement time based on the measurements (assuming that enough time steps remain to steer all controllable states toward their desired final values). In this section, it is shown that, even under the ON–OFF input constraints the overall optimization can be separated into simpler optimization problems, provided some assumptions taken to do

so are verified afterward. Key results are derived for a single measurement case, then expanded to multiple measurements.

A. Single Measurement

In this case, the optimization problem is to determine the minimum of $E[(x(N)-x_d)'(x(N)-x_d)]$ for a single measurement and corresponding measurement time T_l and update time T_u.

The final states can be written as

$$x(N) = A^{N-T_u}x(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}(Bu(i) + B_w w(i)). \tag{9}$$

This can be modified by introducing the estimates at T_u to

$$x(N) = A^{N-T_u}\hat{x}(T_u) + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i). \tag{10}$$

Assuming that there exists $u(i), i = T_u \ldots N-1$ such that

$$A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) \approx x_d \tag{11}$$

or in other words

$$E\left(\left\| A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) - x_d \right\|^2\right) \ll E[(x(N)-x_d)'(x(N)-x_d)] \tag{12}$$

then (10) can be rewritten as

$$x(N) = x_d + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i). \tag{13}$$

It is critical to note that under this solution strategy, (12) determines whether this simplifying approach is valid; this assumption will be revisited in the overall solution procedure in Section IV. It is also worth noting that under ON–OFF control constraints, the actuation error term in (12) is nondecreasing, as each additional time step provides an additional potential adjustment to the input that may cancel out additional deviation from the final desired states based on knowledge at T_u. Thus, given a selected measurement time T_l, T_u would ideally be set equal to T_l. However, in low-power control conditions, computation speed is generally limited, so it is often convenient to update inputs after some delay T_d from the measurement, such that $T_u = T_l + T_d$, and thus the following analysis allows T_u to vary from T_l.

So long as (12) holds true, utilizing intermittent Kalman filter equations the objective function can be written as a function of measurement time T_l alone, separate from the update time T_u, as shown in Appendix A. Hence, for single measurement case, the measurement time T_l can be found by solving the following simpler optimization problem:

$$\min_{T_l}\left[\text{trace}\left\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i} - A^{N-T_l}P_{T_l}C'[CP_{T_l}C'+R]^{-1}CP_{T_l}A'^{N-T_l}\right\}\right] \tag{14}$$

where P_{T_l} is an intermittent Kalman filter error covariance. The quantity in (14) represents the expected value of the minimum error if input constraints have negligible effect on final error, or in other words if the desired state is reachable within some error bound as of the update time. Furthermore, once the measurement time is found, the input sequence can be updated by solving an integer programming problem at any time between T_l and the latest time that the assumption (12) is satisfied within a desired level of accuracy. This step is explained in detail in Section IV.

B. Multiple Measurements

Consider the case with two measurements taken at T_{l1} and $T_{l2} > T_{l1}$. To utilize the information gathered from the second measurement, input must be updated at or after T_{l2}, letting this time be $T_u \geq T_{l2}$.

Using intermittent Kalman filter equations, as derived in Appendix B, the optimization can be reduced to

$$\min_{T_{l1},T_{l2}}\left[\text{trace}\left\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i} - A^{N-T_{l1}}P_{T_{l1}}C'[CP_{T_{l1}}C'+R]^{-1}CP_{T_{l1}}A'^{N-T_{l1}} - A^{N-T_{l2}}P_{T_{l2}}C'[CP_{T_{l2}}C'+R]^{-1}CP_{T_{l2}}A'^{N-T_{l2}}\right\}\right]. \tag{15}$$

Extending this result to a general multiple measurements case where measurements are done at $T_{l1}, T_{l2} \ldots T_{la}$, the times can be found by solving the optimization problem

$$\min_{T_{l1},T_{l2}\ldots T_{la}}\left[\text{trace}\left\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i} - \sum_{i=1}^{a} A^{N-T_{li}}P_{T_{li}}C'[CP_{T_{li}}C'+R]^{-1}CP_{T_{li}}A'^{N-T_{li}}\right\}\right]. \tag{16}$$

When the assumption made earlier is not satisfied, additional updates should be done. Consider the case where b number of updates are made at $T_{u1}, T_{u2} \ldots T_{ub}$, $T_{ub} \geq T_{la}$, with the modified assumption that $E[\|A^{N-T_{ub}}\hat{x}(T_{ub}) + \sum_{i=T_{ub}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2] \ll E[(x(N)-x_d)'(x(N)-x_d)]$. If the modified assumption holds true, again the optimization problem reduces to finding the set of measurement times without regard to update time as expressed in (16).

IV. Solution Strategy

Consider a controllable and observable system with no input constraints. In order to achieve a desired set of states at the

end of an optimization horizon N, measurements need to be taken at least n steps before the end of the optimization horizon, where n is the order of the system. This will give enough time to choose the inputs to drive the states to the desired set. The previous scenario occurs when there is complete measurement of the states. However, when measurements are partial or if there is measurement noise, states must be estimated. Hence, the measurement time could be different from $N-n$.

Similarly, if the system is completely controllable and if there are no input constraints, theoretically it is possible to select a combination of inputs starting at $N-n$ to drive the states to the desired states. However, if there are input constraints, the decision to update the inputs generally needs to be made at an earlier time instant. In the current problem, since the inputs are ON–OFF, their selection is made by solving an integer programming problem. This makes the problem even harder due to the nondeterministic polynomial-time hard (NP)-hard nature of the integer programming problems.

Intuitively, in a single measurement case, one approach is to find the latest time for updating the input sequence with minimal cost and determine the measurement time which minimizes the state covariance at the given update time. However, this proved to be significantly suboptimal. As will be shown [in (26), derived in the Appendix], the measurement time needs to be selected to minimize the trace of state covariance matrix at the end of the optimization horizon, not at the update time. This outcome will also be illustrated in the next section by case study 2, using a brute force solution.

Step 1: Open-Loop Optimal Controller

As noted in Section II, the first step in the solution is to find an optimal switching sequence which minimizes the error in the final states with respect to a set of desired states when disturbances are absent. The objective function in this case can be written as

$$\left[\sum_{i=0}^{N-1} A^{N-1-i}Bu(i)-x_d\right]'\left[\sum_{i=0}^{N-1} A^{N-1-i}Bu(i)-x_d\right] \tag{17}$$

where $u(i)\in 0,1$ are binary. Techniques for solving this problem were described in [1].

Step 2: Selection of Measurement Times

Using the result from the previous section, measurements times are optimized under the assumptions of sufficient time for updating being available. The measurement times $T_{l1}, T_{l2}\ldots T_{la}$ can be obtained by solving the optimization problem

$$\min_{T_{l1},T_{l2}\ldots T_{la}}\left[\text{trace}\left\{A^N P_0 A'^N+\sum_{i=0}^{N-1} A^{N-1-i}B_w Q B_w' A'^{N-1-i}\right.\right.$$
$$\left.\left.-\sum_{i=1}^{a} A^{N-T_{li}}P_{T_{li}}C'[CP_{T_{li}}C'+R]^{-1}CP_{T_{li}}A'^{N-T_{li}}\right\}\right]. \tag{18}$$

Step 3: Assumption Verification

Case 1 (Single Update): However, the assumption must be verified for the last measurement time to complete the procedure, so that an update time ($T_u=T_l$ or $T_u=T_l+T_d$, depending on use of delay) can be obtained. If the assumption is satisfied for the last measurement time, only one input update is required. Since the mean and covariance of state at the final measurement time can be estimated easily, the assumption for the single update case can be checked easily using the following procedure.

The mean of the state estimate at T_{la} can be written as

$$E\{\hat{x}(T_{la})\}=\sum_{i=0}^{T_{la}-1} A^{T_{la}-1-i}Bu(i) \tag{19}$$

where $u(i)$ in the previous equation is the input sequence obtained from the initial open-loop optimization. The covariance of the states at T_{la} based on measurements at $T_{l1}, T_{l2}\ldots T_{la}$ can be found using the recurring intermittent Kalman filter equations; covariance of states at T_{la} after measurement ($P_{T_{la}/T_{la}}$) can be written in terms of covariance before measurement ($P_{T_{la}}$) as

$$P_{T_{la}/T_{la}}=P_{T_{la}}-P_{T_{la}}C'[CP_{T_{la}}C'+R]^{-1}CP_{T_{la}} \tag{20}$$

and $P_{T_{la}}$ can be written in terms of covariance of states after previous measurement at PT_{la-1}/T_{la-1} as

$$P_{T_{la}}=A^{T_{la}-T_{la-1}}P_{T_{la-1}}A'^{T_{la}-T_{la-1}}+\sum_{i=0}^{T_{la}-T_{la-1}-1} A^i BQB'A'^i. \tag{21}$$

This can be extended to the initial state covariance P_0. A sample set of Gaussian pseudorandom variables with the given mean and covariance are generated for $\hat{x}(T_{la})$. For each set of these random variables, corresponding input sequences minimizing $\|A^{N-T_{la}}\hat{x}(T_{la})+\sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i)-x_d\|^2$ can be determined by solving an integer programming problem. In order to check the assumption, $E(\|A^{N-T_{la}}\hat{x}(T_{la})+\sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i)-x_d\|^2)$ is evaluated at the end to compare with (18).

Sometimes, it is desirable to determine the latest update time satisfying the assumption since it gives more time to compute the updated sequence hence allowing the use of a processor with a slower frequency. This is achieved by separating the restrictions posed by ON–OFF constraints from the estimation errors. For that, the error function was evaluated when full state information is available (output matrix C=I and no noise) and compared that with the estimation errors obtained from (18). The update can be done at any time that the input constraints don not hinder the ability to reduce the error function to a negligible level, compared to the estimation errors. A result obtained from such a comparison for the parameters given in the case studies is shown in Fig. 3. In this example, the best measurement time is 16 and update can be done as late as 20 without incurring significant disadvantages to final controller cost.

Case 2 (Multiple Updates): When the assumption is not satisfied at the final update time as discussed in the last section, updating more than once can reduce the value of $E(\|A^{N-T_{la}}\hat{x}(T_{la})+\sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i)-x_d\|^2)$ and hence multiple updates can

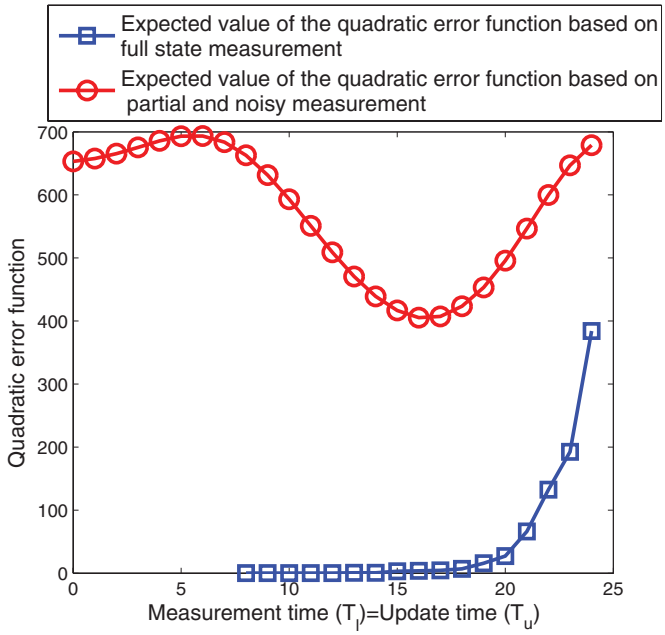

Fig. 3. Comparison of error functions with full and partial measurements.

be useful. In the multiple update case, since integer programming is used to obtain the input sequence after the first update, the probability distribution of $\hat{x}(T_{la})$ cannot be determined analytically. Instead, a Monte Carlo method is used to obtain probability distributions.

The alternate update time selection method in this case begins by updating inputs at all the measurement times in the Monte Carlo simulation and checking the assumption for the final update time as was done case 1. If the assumption is satisfied, the optimal measurement times obtained from the reduced optimization remains optimal. In this case, there is a possibility of reducing the number of updates which is beneficial in reducing the number of online computations. This can be done by using the following algorithm.

To reduce the number of updates, if possible, starting from the final measurement time find the latest time instant T_{u1} for which $E(\|A^{N-T_{u1}}\hat{x}(T_{u1}) + \sum_{i=T_{u1}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2) \ll E[(x(N)-x_d)'(x(N)-x_d)]$ using Monte Carlo methods. Let $T_{lm} \leq T_{u1} \leq T_{ln}$. Now, evaluate the assumption incorporating the new update again on the final measurement time. The new update reduces the value of $E(\|A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2)$; if the reduction is sufficient to satisfy the assumption, one of the new optimal time updates is T_{u1} associated with T_{la}. If the assumption is not satisfied, again the Monte Carlo method needs to be applied to find the latest update time between T_{ln} and T_{la}. Repeating this method until the assumption is satisfied at the final measurement time gives a reduced number of updates if possible.

The separation of an optimizing controller and sensor scheduling is not possible if the assumption is not satisfied at the final measurement time even after updating at all the measuring time instants. In this case in order to obtain a set of optimal measurement and update times, combined optimization needs to be done using a sufficiently large sample set of random variables via Monte Carlo methods.

Computational Advantage

In a single measurement and update case, in order to determine T_l and T_u using brute force optimization, ideally all feasible combinations would be considered, where $T_l \in [0, N-1]$ and $T_u \in [T_l, N-1]$, though if a fixed desirable delay between T_l and T_u is selected beforehand, the number of combinations is reduced. For each of these combinations, an integer programming problem of size $N - T_u$ needs to be solved. Moreover, each of these integer programming problems needs to be done over a number of sets (say S total sets) of random variables to determine the combination of T_l and T_u that minimizes the mean of $(\hat{x}(N) - x_d)'(\hat{x}(N) - x_d)$, via Monte Carlo methods. Although this can be done offline, integer programming problems are computationally hard to solve, belonging to the class of optimization problems known as NP-hard problems. This makes solution of integer programming problems very expensive for a larger problem size. In the proposed strategy, T_l can be evaluated algebraically without solving any integer programming problem. In order to find T_u, integer programming problems of size $N - T_u$ may only need to be solved for a single T_u, to verify that assumptions on final state reachability are met, though simulations at additional time steps may be performed if delay between T_u and T_l is flexible, and necessary to perform through the entire optimization horizon if the assumptions are not met. To summarize, by the proposed method, as few as S number of integer programming problems often have to be solved compared to $N \times S$ in the brute force method. In the multiple measurement case, the computational advantage is twofold. By the strategy proposed, it is possible to reduce the number of updates without affecting the overall performance. This will reduce the memory storage and/or computational capacity required on board. Moreover, the proposed strategy gives a systematic way of finding each update time one by one compared to solving a large optimization problem involving all the update times, measurement times, and binary inputs.

V. Case Studies

The following second-order system was identified for the microrobotic leg shown in Fig. 2:

$$A = \begin{bmatrix} 0.984 & 9.795 \times 10^{-5} \\ -270.343 & 0.955 \end{bmatrix} \quad B = \begin{bmatrix} 1.273 \times 10^{-4} \\ 2.527 \end{bmatrix}.$$

The sampling time used for discretization was 0.1 mS and input voltage at the ON state was set to be 20 V. The range of motion for this voltage level was 0.2 rad. For illustrative examples, the objective is set to reach in the neighborhood of (0.1 rad, 0 rad/s) at $N = 25$th time step. In this microdevice, having relatively high natural frequency, variation in angular velocity is numerically large compared to that of angular displacement. In order to normalize the error penalty in the cost function (8), the first state is weighted by the natural frequency (5.3×10^4 rad/s) of the system. Thus, the costs shown throughout this paper have units equivalent to $[\text{rad/s}]^2$. Disturbances are assumed to

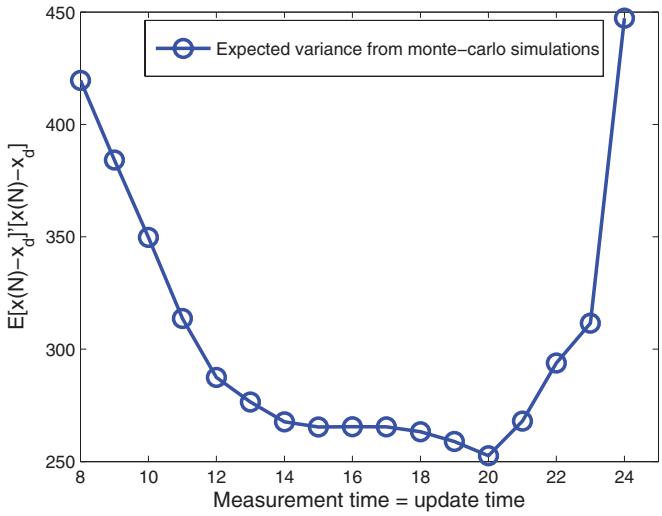

Fig. 4. Variation in the objective function with respect to measurement time for the full state measurements.

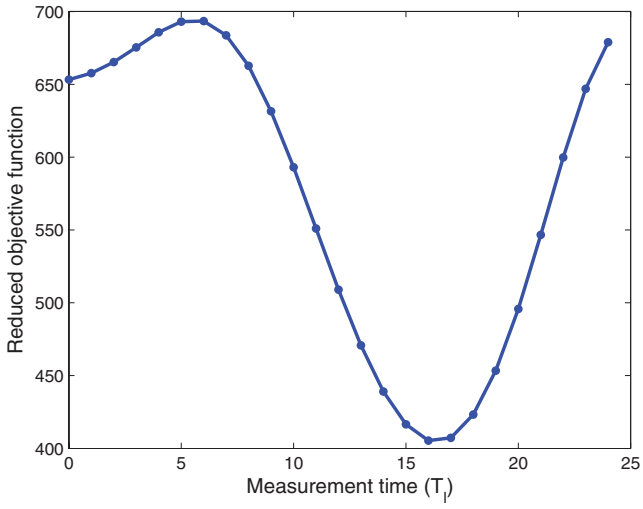

Fig. 5. Variation of reduced objective function with respect to measurement time for the partial state measurement case.

be normally distributed with standard deviation of $\sqrt{10}$ V and initial conditions are also set to be normally distributed with standard deviations 10^{-1} and 10, respectively. The input disturbance strength is equivalent to 15% variation in the voltage. This is somewhat arbitrary for illustration of the approach, but in part reflects substantial irregular ripple voltage in voltage converters or voltage regulators designed to operate at very low-power throughput without excessive sacrifice in efficiency.

Throughout the simulational study, the integer programming problems were solved using CPLEX.

A. *Full Measurement With No Noise*

First a simple case is considered. The output matrix C is assumed to be the identity matrix and it is assumed that there is no noise present in the measurements. Also, the input sequence is updated at the measurement time itself (no delay). Substituting $C = I$ and $R = 0$ into (14), the objective function reduces to the following:

$$\min_{T_l}\left[\text{trace}\left\{A^N P_0 A'^N + \sum_{i=0}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i} - A^{N-T_l} P_{T_l} A'^{N-T_l}\right\}\right] = \min_{T_l}\left[\text{trace}\left\{\sum_{i=T_l}^{N-1} A^{N-1-i} B_w Q B_w' A'^{N-1-i}\right\}\right]. \quad (22)$$

This equation suggests that the best option is to measure at $N-1$ to capture the effect of disturbances on the last time period. However, since there are no choices left to update inputs at this instant, the assumption that $A^{N-T_u}\hat{x}(T_u) + \sum_{i=T_u}^{N-1} A^{N-1-i} B u(i) \approx x_d$ becomes invalid. Hence, measurement needs to be done at an earlier time step when there are enough input choices left to drive the final states close to x_d. This is verified by the simulation result given in Fig. 4. The variation in the objective function for a measurement time range of steps 8 to 24 is shown in the figure. In this particular case, variation in the objective function for $T_l = 14$ to 21 is small and any time in this range is acceptable. After 21st time step, although the estimation error of the final states is smaller, the input choices available are not sufficient to reduce the final error to the desired range.

To put the numerical cost from this example into context, the minimum cost of 250 is equivalent to an average velocity error of approximately 15 rad/s at the conclusion of motion, as compared to a maximum angular velocity of the system of approximately 126 rad/s. Without any feedback, the average velocity error would have been nearly 26 rad/s, while the best case final velocity (had there been no disturbance) is less than 2 rad/s. Error in the position state is typically smaller relative to maximum displacement, with average error under feedback of approximately 0.01 rad, or 10% of the final position amplitude.

B. *Partial Measurement With Noise*

In this case, only the first state (angular displacement) is measured and hence the output matrix $C = [1 \quad 0]$. The objective, disturbance levels and initial conditions are set to be the same as the previous case. The measurement noise levels assumed to have a standard deviation of 0.01 rad.

1) Single Measurement and Single Update: First, to illustrate the selection of the best measurement time, (27) is plotted for $T_l = [0, N-1]$ in Fig. 5. The reduced objective function achieves a minimum for $T_l = 16$. In order to verify the assumption from (12), $E(\|A^{N-T_l}\hat{x}(T_l) + \sum_{i=T_l}^{N-1} A^{N-1-i} B u(i) - x_d\|^2)$ is evaluated at $T_l = 16$ using a Monte Carlo simulation with a sample size of 1000. Error from the update is on the order of 2 (2.346 for the used sample set), very small compared to cost from (27) given in Fig. 5, thus confirming the validity of the assumption in this case.

In order to further verify this result, a comparison with the results obtained from the brute force method is also done. A

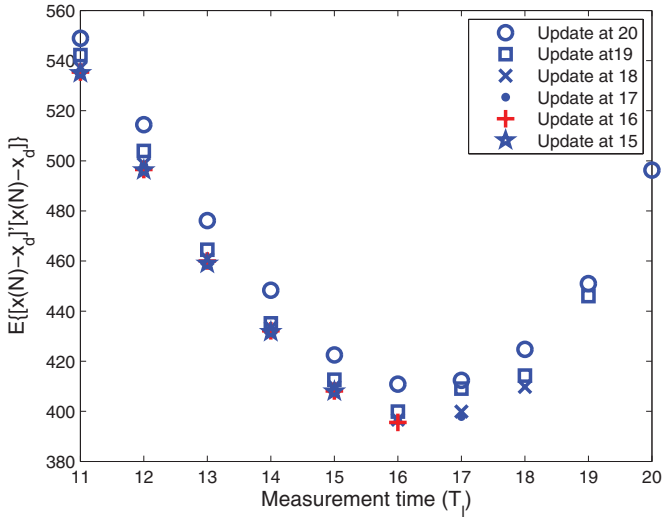

Fig. 6. Variation in the objective function with respect to measurement time and update using complete enumeration time for the partial state measurement case.

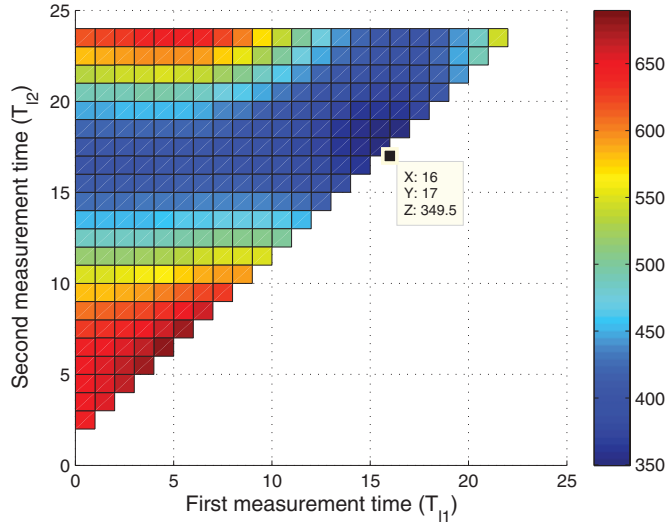

Fig. 7. Variation in objective function with two measurement times.

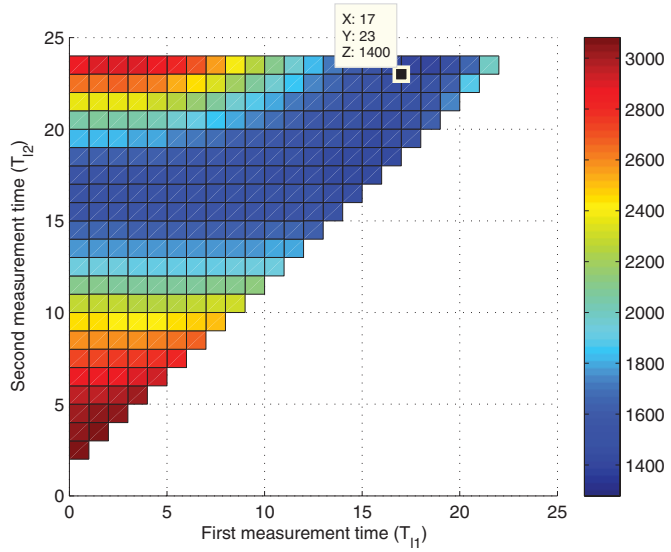

Fig. 8. Variation in objective function with two measurement times with higher disturbance level.

complete enumeration of all possible combinations of measurement and update times were done for a sample set of size 1000 pseudorandom variables. Results are shown in Fig. 6, which confirms that the best measurement time is 16, as well as the fact that update can be done anytime between the 16 and 19th time steps with $<1\%$ error introduced to the final cost. In other words, up to three step delays for computation are tolerable, and evaluation of this fact using the reduced computation sequence would require Monte Carlo simulations only for time steps 16 to 20.

2) Multiple Measurement and Single Update: When the latest update time is not too late, the quadratic error function can be reduced if more energy is spent on sensors, in other words by taking more measurements. For example, in this case, two measurements are taken resulting in a better performance for the same disturbance and noise levels. In order to find the best measurement times T_{l1} and T_{l2}, (15) is plotted in Fig. 8. It was found that the best measurement times are at 16 and 17 and the corresponding error function is 349.5. As the next step described in the solution strategy, the assumption was verified for an update time $T_u = 17$. Also, from the Monte Carlo method, the objective function was found to be 350.3 verifying the result.

3) Multiple Measurement and Multiple Update: When disturbance levels are higher, $E(\|A^{N-T_{la}}\hat{x}(T_{la}) + \sum_{i=T_{la}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2)$ becomes larger for a relatively late last measurement time and it becomes beneficial to update more than once. In this scenario, the disturbance standard deviation was chosen to be $Q = 50$ and all the other parameters kept the same. Variation in (15) is plotted in Fig. 7. Measurements are taken twice over the entire optimization horizon. The best measurement times from the reduced optimization given in (15) are $T_{l1} = 18$ and $T_{l2} = 24$, however because these times leave only a single time step for the second update to take effect measurement times were indexed forward one time step to $T_{l1} = 17$ and $T_{l2} = 23$. The expected value of the objective function from (15) is 1400 as shown in the figure. The assumption was checked using the Monte Carlo method updating at $T_{l2} = 23$ only and it was found that $E(\|A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2) = 325.8$ resulting in an objective function value of 1802.3. In order to get closer to the expected analytical solution, two updates were then done at $T_{u1} = 17$ and $T_{u2} = 23$, resulting in a better value of the objective function at 1565.5. The reason for the difference in this value from the analytical solution can be accounted for by the large error (for this scenario, though small compared to single update scenario) in $E(\|A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d\|^2) = 138.0965$.

The point of this example was to verify that in multiple measurement cases where the assumption that the state is reachable from the final update is not satisfied at the last measurement time, it is beneficial to update more than once. Also, when the assumption is not satisfied, the difference

TABLE I
COMPILATION OF RESULTS

	Expected error from reduced optimization	Error shown in Monte Carlo method	Number of integer programming problems solved	
			brute force method	Reduced optimization
Open-loop	695.4	697.9	1	1
SMSU	405.4	404.1	25001	1001
TMSU	349.5	350.3	300001	1001
TMSU (higher disturbance)	1400	1802.3	300001	1001
TMTU (higher disturbance)	1400	1565.5	324001	2001

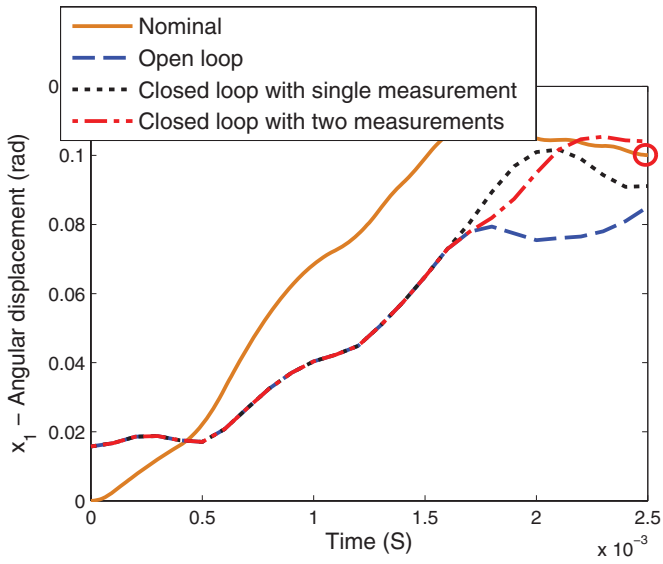

Fig. 9. Time response comparison of first state showing the effect of feedback.

between the value calculated by (15) and $E(\text{trace}\{[x(N) - x_d]'[x(N) - x_d]\})$ is due to the variance of $A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d$ and cross covariance between $A^{N-T_{l2}}\hat{x}(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}Bu(i) - x_d$ and (15). Since the calculations of these terms involves integer programming, evaluation of them analytically is not in general possible.

The results of all these case studies are compiled in Table I. The acronyms used in the table are SMSU for single measurement single update; TMSU for two measurement single update; TMTU for two measurement two update. The number of integer programming problems used to solve each case is also tabulated. Sample time responses are shown in Figs. 9 and 10, for the trajectories of the first and second states, respectively. This set of responses corresponds to one of the worst open-loop performances in Monte Carlo simulations. The solid line represents the nominal performance in the absence of disturbances and initial state uncertainties. As expected, after 25 time steps (at

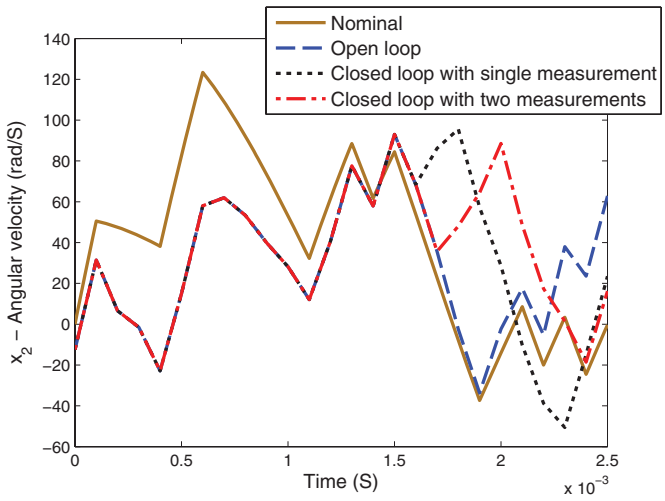

Fig. 10. Time response comparison of second state showing the effect of feedback.

2.5 ms) nominal states reach very close to (0.1 rad, 0 rad/s.). The dashed lines show the perturbed open-loop performance, with the deviation in the final states due to random disturbances and initial conditions, using the nominal input sequence. The dotted line represents closed-loop performance with a single measurement and the dashed-dot line shows closed-loop performance with two measurements and a single update.

Both of the closed-loop performances shown in Figs. 9 and 10 show improvement over the open-loop performance. First, note that variation in velocity is much larger than that in displacement numerically. This behavior helped prompt the decision to penalize velocity more heavily than displacement, and compute costs in $[\text{rad/s}]^2$ for the goal of these case studies to bring an actuator to a specified position with near-zero velocity at the end of the response. In the open-loop response, the first state reaches only 0.85 rad at the desired time showing an error of 15%. On the other hand, the single measurement case settles at 0.91 rad and two measurements case reaches at 1.04 rad, only 9% and 4% error. For second state, the open-loop response reaches 62.85 rad/s, single measurement arrives at 23.47 rad/s and the two measurement case settles at 16.22 rad/s. Closed-loop performance did show improvement in nearly all Monte Carlo simulations, though the difference between single and multiple measurements is generally not as significant as for this specific simulation.

4) Implementation of Feedback: Given limited computation and memory capacity of autonomous microsystems, it is proposed that to implement intermittent feedback, a finite library of ON–OFF update sequences be stored onboard. For example, consider the case discussed earlier where measurement is done at the 16th time step, while update is done at the 19th time step to allow for a plausible computation time in selecting and loading the best sequence. From the supporting Monte Carlo analysis, it is seen that only 31 ON–OFF sequences of only 6 bits each need to be stored to cover the output variation seen, that is, out of 2^6, or 64, possible sequences from that time forward. Furthermore, smaller numbers of sequences can be stored and

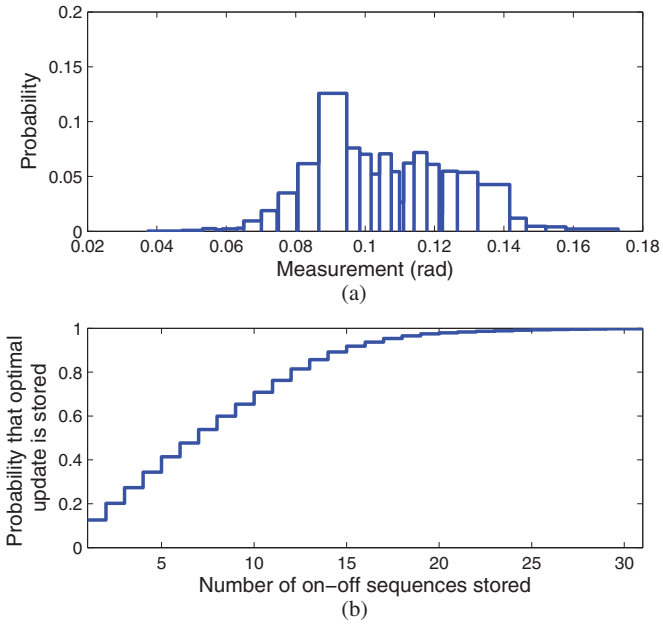

Fig. 11. (a) Partition of measurement space based on ON–OFF updates made at T = 19. (b) Cumulative probability that an optimal update is available for a given measurement versus number of update sequences stored.

still optimally cover substantial fractions of the feasible space. The probability of a measurement being in one of the subranges of measured displacement determining the input sequence to apply is shown in Fig. 11(a). Naturally, the partition locations must also be stored in memory, adding additional, but not excessive, memory requirements. In cases when on-board memory is further limited, only a subset of the update sequences may be desired to be stored. In Fig. 11(b), probability of an optimal sequence being present when a certain number of sequences are stored is given.

To further reduce energy consumption, the objective function for initial optimization (see step 1 in Section IV) can be modified to minimize switching cost associated with charging the piezoelectric actuator each time, into

$$J_C = \frac{1}{2} C u_{\max}^2 \left(\sum_{k=0}^{n} (u(k) - u(k-1))^2 \right) \qquad (23)$$

where C is the capacitance of the piezoelectric actuator array [see Fig. 1(b)] and $u_{\max}$ is the "ON" state voltage of the ON–OFF controller. The binary variable $u(k) = 1$ corresponds to the "ON" state of the controller and $u(k) = 0$ corresponds to the "OFF" state. Since the energy consumed for one charging of the actuator is $\frac{1}{2} C u_{\max}^2$ and nearly all energy used by a piezoelectric actuator is consumed when it charges, this objective acts to minimize the number of chargings and dischargings in the actuator array. The constraints for the optimization are the deterministic part of the dynamics equations given in (1). A Monte Carlo study was done to document the advantages of this modification on the energy consumption. The average number was seen to decrease from an average of 6 from the original optimization to an average of 2 for the combined optimization with energy minimization, although there is a 5% increase in the error values.

VI. Conclusion

Use of an energy consuming sensor sparingly with an ON–OFF controller is studied here, where the sensor is used only for a limited number of sampling instants over a finite optimization horizon. Using state estimates from these measurements, the optimal open-loop ON–OFF input sequence driving states of the system to desired final values is updated at one or more update times following measurements. A three-step near-optimal strategy for selecting measurement and update times is proposed. First, the open-loop input sequence without any measurements is obtained. Second, measurement times are selected independent of update times under the assumption that updating inputs after the measurement time or some desired delay later will approximately cancel error in states projected from the measurements. Third, a Monte Carlo simulation of response to stochastic variation for the proposed update time is performed, to verify that assumption's validity. Procedures for adjusting measurement and update times if the assumption fails, for eliminating unnecessary updates having negligible impact on performance, and for incorporating actuator power consumption are proposed. Several case studies for a microrobotic leg joint are examined in simulation to explore the proposed controller design effectiveness.

The procedure described has potential usefulness for very low power control systems, where analog inputs and sensor measurements are expensive from an energy standpoint, and rapid computation or data lookup are not necessarily available. However, it is worth noting that the proposed results, even accounting for actuator energy consumption, are not guaranteed to be optimal for overall energy consumption. Rather, limits are effectively placed on energy use for sensing and on computation time, and near-optimal energy usage just for actuation is obtained under these constraints. A second limitation is that the current approach focuses on Gaussian disturbances at each time step, which is a common approach to a controller design and can be adapted to some other situations, but may not always best describe disturbances experienced by microsystems for which this control approach is primarily intended. Meanwhile, the approach provides decent computational benefits over baseline optimization procedures, such as full evaluation by Monte Carlo methods. Online computation or data lookup in the low-power system is also kept to a low level. Possible future enhancements to the control approach include accounting for system model variation and evaluating minimal total energy requirements, including sensor and computation costs, for various performance levels in terms of final state error.

VII. Discussion

Many earlier studies in sensor scheduling have been done to decide on the frequency at which measurements need to be taken to achieve certain feedback controller performance level. In this paper, the objective was to find the best time instants at which measurements may be taken for adjustment of the input

sequence of an ON–OFF controller. This approach is prompted by the observation that sensing circuitry often consumes power at the same order of magnitude or larger than miniaturized actuators. Using limited numbers of measurements over the course of motion can potentially keep overall system power consumption low, while providing a substantial portion of the benefits of real-time feedback control.

Practical limitations of any specific sensor to be used with this control scheme may lead to certain modifications. For example, there are various ways to modify the strategy to account for a sensor with a warm-up period. If a motion is to be repeated many times at a consistent frequency, as is the case for walking with a microrobotic leg joint, the sensor may sample at a consistent frequency (once per motion) and the first few motions may simply have larger errors. For unrepeated motions, in the multiple measurement case, early measurements might be assumed to be less reliable than later ones; noise levels used to calculated Kalman Filter gains can be modified for time-varying noise intensity. Additional measurements might also be added, to help ensure that the measurement at the optimal time point occurs when the sensor is operating at its best performance, although this will entail additional energy use. Alternatively, if the warm-up period characteristics of the sensor are consistent, the estimation equations can be recalibrated to include predictions from the known sensor behavior in that period. Similarly, if the sensor itself has known sensor dynamics, these can be included in the dynamic model used by the optimization algorithm. Or, since the measurement time instants are predetermined, if only the duration of dynamic sensor behavior is known, the sensors can be turned ON early enough so that distortion of the measurement is minimized, though perhaps not entirely eliminated.

As noted before, a partial limitation of this approach is that energy consumed by the sensors is not directly included in the objective function used in this paper. Instead, implications for power usage must be analyzed in terms of specific gains in performance versus sensing and actuation energy costs for a given system. For the case study in the paper, energy analysis may begin with the open-loop and optimal closed-loop costs shown in Table I, which are weighted sum of squares of errors in final states. For a single measurement, square root of cost, equivalent to velocity error in rad/s, is reduced by about 25% for the single measurement case and by 30% for the two measurement case; further measurements show further diminishing returns due to the input constraints imposed by ON–OFF control. Thus, the most direct advantage of the proposed method is to reduce output error under a given sensing power budget, using significantly less energy than a conventional feedback controller with frequent sensing.

However, there are also ways to directly compare sensor and actuator energy usage. For example, with average equivalent velocity and positioning errors reduced by 25% from open-loop operation, the need for corrective action (i.e., additional motions of a robot leg to make up lost distance traveled) was reduced by about 4–10% (not the full 25% since an open-loop error is finite and not all motions require correction). Thus, required actuator energy is reduced by 16–40 nJ for the sample 1-nF actuators. This is less than the 500-nJ sensing energy estimate, so with current technology the proposed control scheme is best for improving performance under an energy limit, rather than reducing net energy use. However, scenarios where total energy consumption is reduced are easily anticipated. One scenario is use of larger actuators (with capacitance of 30 nF or more), and the other is for lower power sensing circuits, on the order of 50 nJ, with one sensor satisfying this requirement recently reported in the literature [8]. A similar analysis can be performed for increasing number of sensor measurements to find the best case combination of performance and power consumption; the single measurement approach is judged to be the best for the current microrobotic leg joint case study by the authors.

To conclude, the procedure described in this paper provides an efficient and highly effective method for designing the sampling scheme of ON–OFF switching controllers with limited numbers of feedback measurements. Both ON–OFF inputs and infrequent sensing permit significant reductions in power consumption for feedback control of capacitive microactuators compared to conventional control schemes. While the resulting input constraints and measurement uncertainty reduce motion controller accuracy, the approach can be a powerful method for meeting strict energy budgets, such as those faced by autonomous microsystems. A case study was provided to illustrate the approach for a leg control in an autonomous microrobot, and implications for motion accuracy, power consumption, and processing requirements have been noted. Future improvements to the system may come from synthesis with more advanced driving and sensing circuitry, or by improving robustness to model error or sensor timing discrepancies.

APPENDIX A
SINGLE MEASUREMENT CASE

Derivation of (14) under the assumption (12) is described in this section. Tracking system states at the final time $T(N)$ back to the measurement time T_l, (13) can be described using Kalman filter gain K_l as in (7)

$$\begin{aligned} x(N) &= x_d + A^{N-T_l}(x(T_l) - \hat{x}_{T_l/T_l}) + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i) \\ &= x_d + A^{N-T_l}[x(T_l) - (\hat{x}(T_l) + K_l(Cx(T_l) \\ &\quad - C\hat{x}(T_l) + v(T_l)))] + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i) \\ &= x_d + A^{N-T_l}[(I - K_l C)(x(T_l) - \hat{x}(T_l))] \\ &\quad - A^{N-T_l} K_l v(T_l) + \sum_{i=T_l}^{N-1} A^{N-1-i} B_w w(i). \end{aligned} \tag{24}$$

The error in estimation of states $x(T_l) - \hat{x}(T_l)$ is due to the randomness of the initial states and the normal disturbances

until $T_l - 1$. Thus, (24) can be rewritten as

$$x(N) = x_d + A^{N-T_l}\Bigg[(I - K_lC)(A^{T_l}x(0) + \sum_{i=0}^{T_l-1} A^{T_l-1-i}B_w w(i))\Bigg] - A^{N-T_l}K_l v(T_l) + \sum_{i=T_l}^{N-1} A^{N-1-i}B_w w(i). \quad (25)$$

Hence, $E\{[x(N) - x_d]'[x(N) - x_d]\}$ can be written as

$$E\{[x(N) - x_d]'[x(N) - x_d]\} = \text{trace}\Bigg\{A^{N-T_l}(I - K_lC)P_{T_l}(I - K_lC)'A'^{N-T_l} + A^{N-T_l}K_lRK_l'A'^{N-T_l} + \sum_{i=T_l}^{N-1} A^{N-1-i}B_wQB_w'A'^{N-1-i}\Bigg\} \quad (26)$$

$$= \text{trace}\Bigg\{A^NP_0A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_wQB_w'A'^{N-1-i} - A^{N-T_l}P_{T_l}C'[CP_{T_l}C' + R]^{-1}CP_{T_l}A'^{N-T_l}\Bigg\}. \quad (27)$$

APPENDIX B
MULTIPLE MEASUREMENTS CASE

This section describes the derivation of (15) for two or more measurements. The final state $x(N)$ can be written based on the states and the expected values of states at update time $x(T_u)$ as

$$x(N) = A^{N-T_u}\hat{x}(T_u) + A^{N-T_u}(x(T_u) - \hat{x}(T_u)) + \sum_{i=T_u}^{N-1} A^{N-1-i}Bu(i) + \sum_{i=T_u}^{N-1} A^{N-1-i}B_w w(i). \quad (28)$$

Under the assumption (12), tracking the states back to the measuring time T_{l2}, the equation can be modified using Kalman filter gain K_{l2} as

$$\begin{aligned} x(N) &= x_d + A^{N-T_{l2}}[x(T_{l2}) - (\hat{x}(T_{l2}) + K_{l2}(Cx(T_{l2}) - C\hat{x}(T_{l2}) + v(T_{l2})))] + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i) \\ &= x_d + A^{N-T_{l2}}[(I - K_{l2}C)(x(T_{l2}) - \hat{x}(T_{l2}))] - A^{N-T_{l2}}K_{l2}v(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i). \end{aligned} \quad (29)$$

This can be traced back to the states at the first measurement time T_{l1} and written as a sum of independent random variables

$$\begin{aligned} x(N) &= x_d + [A^{N-Tl2} - A^{N-Tl2}K_{l2}C][A^{T_{l2}-T_{l1}} - A^{T_{l2}-T_{l1}}K_{l1}C] \times \left(A^{T_{l1}}x(0) + \sum_{i=0}^{T_{l1}-1} A^{T_{l1}-1-i}B_w w(i)\right) \\ &\quad - [A^{N-T_{l2}} - A^{N-T_{l2}}K_{l2}C]A^{T_{l2}-T_{l1}}K_{l1}v(T_{l1}) + [A^{N-Tl2} - A^{N-Tl2}K_{l2}C]\sum_{i=T_{l1}}^{T_{l2}-1} A^{T_{l2}-1-i}B_w w(i) \\ &\quad - A^{N-T_{l2}}K_{l2}v(T_{l2}) + \sum_{i=T_{l2}}^{N-1} A^{N-1-i}B_w w(i). \end{aligned} \quad (30)$$

Hence, $E\{[x(N) - x_d]'[x(N) - x_d]\}$ can be written as

$$E\{[x(N) - x_d]'[x(N) - x_d]\} = \text{trace}\Bigg\{A^NP_0A'^N + \sum_{i=0}^{N-1} A^{N-1-i}B_wQB_w'A'^{N-1-i} - A^{N-T_{l1}}P_{T_{l1}}C'[CP_{T_{l1}}C' + R]^{-1}CP_{T_{l1}}A'^{N-T_{l1}}\Bigg\} - A^{N-T_{l2}}P_{T_{l2}}C'[CP_{T_{l2}}C' + R]^{-1}CP_{T_{l2}}A'^{N-T_{l2}}\}. \quad (31)$$

REFERENCES

[1] B. Edamana, B. Hahn, J. S. Pulskamp, R. G. Polcawich, and K. Oldham, "Modeling and optimal low-power on-off control of thin-film piezoelectric rotational actuators," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 5, pp. 884–896, Oct. 2011.

[2] J. Main, D. Newton, L. Massengill, and E. Garcia, "Efficient power amplifiers for piezoelectric applications," *Smart Mater. Struct.*, vol. 5, no. 3, pp. 766–775, Aug. 1996.

[3] D. Campolo, M. Sitti, and R. S. Fearing, "Efficient charge recovery method for driving piezoelectric actuators with quasi-square waves," *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 50, no. 3, pp. 237–244, Mar. 2003.

[4] K. Oldham, J. Pulskamp, R. Polcawich, and M. Dubey, "Thin-film pzt actuators with extended stroke," *J. Microelectromech. Syst.*, vol. 17, no. 4, pp. 890–899, Aug. 2008.

[5] Analog Devices, Inc. (2007). *Ultra-Low Power, 2-Channel, Capacitance Converter for Proximity Sensing*, 1–28 [Online]. Available: www.analog.com

[6] D. Fang and H. Xie, "A low-power low-noise capacitive sensing amplifier for integrated CMOS-MEMS inertial sensors," presented at the Int. Conf. Circuits, Signals, Syst., Clearwater Beach, FL, 2004.

[7] W. Bracke, P. Merken, R. Puers, and C. Van Hoof, "Ultra low power capacitive sensor interface with smart energy management," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 130–139, Jan. 2007.

[8] J. H.-L. Lu, M. Inerowicz, S. Joo, J.-K. Kwon, and B. Jung, "A low-power, wide-dynamic-range semi-digital universal sensor readout circuit using pulsewidth modulation," *IEEE Sens. J.*, vol. 11, no. 5, pp. 1134–1144, May 2011.

[9] R. Harrison, P. Watkins, R. Kier, R. Lovejoy, D. Black, B. Greger, and F. Solzbacher, "A low-power integrated circuit for wireless 100-electrode neural recording system," *IEEE J. Solid-State Circuits*, vol. 42, no. 1, pp. 123–133, Jan. 2007.

[10] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1453–1464, Sep. 2004.

[11] Y. Mo and B. Sinopoli, "Kalman filtering with intermittent observations: Tail distribution and critical value," *IEEE Trans. Autom. Control*, vol. 57, no. 3, pp. 677–689, Mar. 2012.
[12] P. S. Yong and A. Sahai, "Intermittent kalman filtering: Eigenvalue cycles and nonuniform sampling," in *Proc. Amer. Control Conf.*, San Francisco, CA, Jun. 29–Jul. 1, 2011, pp. 3692–3697.
[13] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. Sastry, "Foundations of control and estimation over lossy networks," *Proc. IEEE*, vol. 95, no. 1, pp. 163–187, Jan. 2007.
[14] M. Epstein, L. Shi, A. Tiwari, and R. M. Murray, "Probabilistic performance of state estimation across a lossy network," *Automatica*, vol. 44, pp. 3046–3053, 2008.
[15] O. C. Imer, S. Yuksel, and T. Basar, "Optimal control of LTI systems over unreliable communication links," *Automatica*, vol. 42, pp. 1429–1439, Sep. 2006.
[16] C. Li and N. Elia, "A sub-optimal sensor scheduling strategy using convex optimization," in *Proc. Amer. Control Conf.*, San Francisco, CA, Jun. 29—Jul. 01, 2011, pp. 3603–3608.
[17] Y. He and E. K. Chong, "Sensor scheduling for target tracking: A monte carlo sampling approach," *Digital Signal Process.*, vol. 16, pp. 533–545, Sep. 2006.
[18] M. Vitus, Z. Wei, A. Abate, J. Hu, and C. Tomlin, "On efficient sensor scheduling for linear dynamical systems," in *Proc. Amer. Control Conf.*, Baltimore, MD, Jun. 30–Jul. 02, 2010, pp. 4833–4838.
[19] E. Iwasa and K. Uchida, "Model predictive sensor scheduling," in *Proc. Int. Conf. Control Autom. Syst.*, 2008, pp. 2260–2265.
[20] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.
[21] D. Chu, T. Chen, and H. Marquez, "Finite horizon robust model predictive control with terminal cost constraints," *Proc. Inst. Elect. Eng.—Control Theory Appl.*, vol. 153, no. 2, pp. 156–166, Mar. 2006.

Biju Edamana received the B.Tech. degree in mechanical engineering from the National Institute of Technology, Calicut, India, in 2004, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Michigan, Ann Arbor, in 2008 and 2012, respectively.

He is currently a Post-Doctoral Research Fellow at the University of Michigan. His current research projects include developing low-energy control strategies for micro-robotics and developing robust controllers for 6DOF piezoelectric actuators for calibrating an inertial measurement unit.

Kenn R. Oldham (M'10) received the B.S. degree in mechanical engineering from Carnegie Mellon University, Pittsburgh, PA, in 2000, and the Ph.D. degree in mechanical engineering from the University of California, Berkeley, in 2006.

He is currently an Assistant Professor with the Department of Mechanical Engineering, University of Michigan, Ann Arbor. His research interests include micro-actuator design and applications, optimal design and control, design for controllability, and efficient sensing and power strategies for MEMS devices.

Prof. Oldham is a member of the American Society of Mechanical Engineers.