2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Stochastic Optimal Control Laws for Cellular Artificial Muscles

Lael Odhner[1], Jun Ueda[1,2], and H. Harry Asada[1]

Abstract— **This paper presents a control architecture for artificial muscle materials such as shape memory alloys and polymer actuators. The active material is broken up into many small independent cells that can be regulated in a binary fashion into ON and OFF states, so that the actuator displacement is determined by the number of ON cells. In order to control the number of cells that contract, a novel closed loop feedback control method is employed. Each cell is given a small stochastic finite state machine that governs its transition between ON and OFF states. A central controller globally varies the probabilistic rate with which all of the cells make state transitions. Using this architecture, actuator displacement can be controlled in a stable, robust fashion. Different feedback laws are compared using the fixed policy value iteration algorithm to calculate expected settling time in response to a step reference. A simple law based on calculating expected future behavior is found to be very close to the optimal law computed using the value iteration algorithm. The performance of the control laws is verified on a 50 cell shape memory alloy cellular actuator.**

I. INTRODUCTION

This paper discusses a control system for artificial muscles, materials which produce strain in response to a chemical, electrical, or thermal stimulus. Artificial muscles are of great interest to the robotics community because they promise truly linear actuation at densities comparable to human muscle. This density is a key requirement for integrating the large number of actuators necessary to perform high-DOF tasks, such as the grasping motions of the human hand [1]. Already, shape memory alloy (SMA) actuators, conducting polymer (CP) actuators, and electrostatically-stricted polymer (ESSP) actuators have demonstrated material properties that compare quite favorably with human muscle [2],[3].

Controllability is a major impediment to the adoption of artificial muscle actuators. The strain-producing physical processes in many artificial muscle materials often exhibit difficult-to-control behaviors such as hysteresis and saturation. Limits on strain are often imposed because of material failure, such as over-oxidation in CP actuators or dielectric breakdown in ESSP actuators. Because of this, it is often difficult to use feedback control laws to control artificial muscle materials. Several researchers have proposed an alternative control architecture for artificial muscles, inspired by the structure of biological muscle[4],[5]. The active material within the actuator is broken into many small cells connected together in serial and parallel networks, like the fibers and

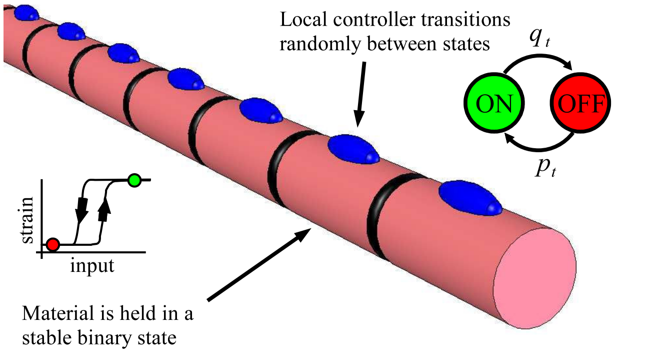

Fig. 1. In a cellular actuator, the artificial muscle material is broken down into many small cells, each with a simple local controller designed to hold the material in a stable relaxed or contracted state. This paper describes a method of activating cells in which the transitions between states occur randomly, with probabilities p_t and q_t determined by a central controller.

motor units in biological muscles. Each cell can be in a contracted, or ON, state or a relaxed, or OFF, state. The summed displacement of all the cells determines the output displacement of the actuator. This architecture decouples the local, non-linear material behavior from the net displacement and compliance produced by the actuator. The control problem remaining is to determine how to recruit the number of cells needed to produce the desired output.

The authors have previously proposed that the recruitment problem can be solved by embedding a small decision maker within each cell that causes the cell to switch randomly between ON and OFF states[6]. This stochastic system does not require that the cells be indivdually addressed, or that the individual cell states be known by a central controller. Instead, the central controller measures the aggregate displacement of the actuator and broadcasts commands to all of the cells, governing the probability with which state transitions are made. Because the actuator displacement is a summation of each cell's randomly determined displacement, the central limit theorem guarantees that the variance of the actuator response will quickly become small as the number of cells becomes large, yielding an almost deterministic response. Previously, a linear feedback law relating the probability of recruitment to the actuator's displacement error has been shown to converge[7]. However, it is clear from simulation and experiment that better performance can be achieved through more thorough modeling of the actuator's stochastic behavior. This paper explores how the choice of a control law impacts the performance of the actuator. In Section II-A, the kinematics governing the motion of the cellular actuator are derived, explaining how the state

This material is based upon work supported by the National Science Foundation under Grant No. IIS-0413242

[1] d'Arbeloff Laboratory for Information Systems and Technology, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, email {lael, uedajun, asada}@mit.edu

[2] Robotics Laboratory, Nara Institute of Science and Technology, Takayama, Ikoma, Nara, 630-0192, Japan

1-4244-0602-1/07/$20.00 ©2007 IEEE.

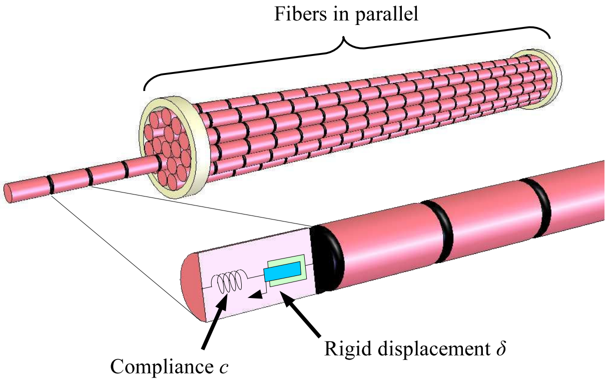

Fig. 2. The actuator is composed of many small cells arranged in a serial-parallel configuration. Each cell is modeled as producing a displacement δ and a compliance c.

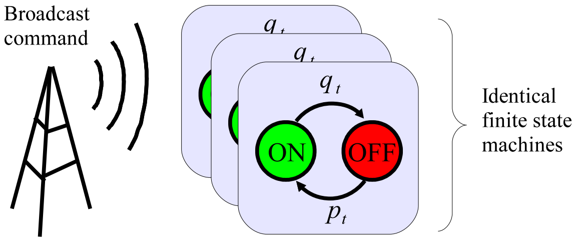

Fig. 3. Each cell responds to the broadcast command, turning ON if it is OFF with probability p_t, and turning OFF if it is ON with probability q_t.

of the actuator can be determined only by the number of ON cells. A Markov model is then derived in Section II-B that describes how cells are recruited to produce motion. Section III then describes how an optimal control law can be computed using the dynamic programming algorithm to minimize the convergence time of the actuator to a step reference input. The value iteration algorithm is also used to evaluate the expected convergence time of the previous linear law and a non-linear law closely approximating the optimal control law. Finally, Section IV confirms the models of Section III with data taken from a fifty cell SMA actuator.

II. THE CELLULAR ACTUATOR ARCHITECTURE

A. A kinematic description

The cellular actuator architecture is broken down in a hierarchy of fibers and cells, shown in Fig. 2. Cells are identical, independently controlled active material units, similar to individual motor units in skeletal muscle[8]. Each cell is modeled as having a small rigid displacement δ in series with a small compliance c. The values of δ and c vary with the state of the cell,

$$\delta = \begin{cases} 0, & state = OFF \\ \eta, & state = ON \end{cases} \tag{1}$$

$$c = \begin{cases} c^{off}, & state = OFF \\ c^{on}, & state = ON \end{cases} \tag{2}$$

These cells are connected together in series to form fibers, as shown in Fig. 2, which are in turn connected in parallel to form the whole actuator. The displacement of the actuator can be predicted through the elastic averaging of each cell's displacement in this serial-parallel kinematic network. We would like to show that the actuator displacement and stiffness can be determined by a simple expression in terms of the total number of ON cells N^{on}, so that this can be used as a state for a state-space model of the actuator. Because a fiber is composed of many compliant cells in series, the net displacement and compliance of each fiber is equal to the sum of the individual cell displacements and compliances within the chain. From (1) and (2), we can see that these in turn reduce to linear functions of N_j^{on}, the number of cells within fiber j of the actuator:

$$\delta_j = \sum_{i=1}^{N_j} \delta_i = \eta N_j^{on} + 0 N_j^{off} = \eta N_j^{on} \tag{3}$$

$$\begin{aligned} C_j &= \textstyle\sum_{i=1}^{N_j} c_i = N_j^{on} c^{on} + N_j^{off} c^{off} \\ &= N_j c^{off} + N_j^{on}(c^{on} - c^{off}) \end{aligned} \tag{4}$$

The parallel combination of fibers to produce the actuator's net output y is slightly problematic. If one fiber is consistently shorter or longer than the others, the contribution of each fiber may not be uniform. Also, the compliance of each fiber can vary with the number of ON cells, so one fiber could potentially be stiffer or more compliant than the others. Fortunately, the state transitions of the cells are random and independent, so there should be no bias in the number of ON cells in each chain. Making these assumptions, y can be approximated as the average displacement of each fiber. For M fibers in parallel,

$$y \approx \eta N^{on}/M + F_d C \tag{5}$$

F_d is a disturbance force, and C is the net compliance of the actuator, which is assumed to be the average compliance of one chain divided by M,

$$C \approx \left[(N/m)c^{off} + (N^{on}/M)(c^{on} - c^{off}) \right] /M \tag{6}$$

Equations (5) and (6) provide a good steady-state approximation for the behavior of the cellular actuator. If the transient response time of the active material is short relative to the time scale of motion, the number of cells ON at time t, N_t^{on}, will uniquely determine the displacement of the actuator.

B. A Markov model of cell recruitment

In order to predict the time evolution of the cellular actuator, some model for predicting the future state $N_{t+\tau}^{on}$ from the present state N_t^{on} must be found. Because of the stochastic nature of the recruitment process, the joint probabilities of each cell responding to a broadcast command can be used to derive a Markov model for $N_{t+\tau}^{on}$ conditioned on the globally broadcast transition probabilities p_t and q_t. This Markov model can then be used to analyze and synthesize closed-loop feedback control laws for the actuator.

The ON or OFF state of each cell is dermined by a small finite state machine within each cell. A state transition graph for the state machine is depicted in Fig. 3. At regular time intervals of τ seconds, each cell makes a random decision to transition from OFF to ON with probability p_t, and from ON to OFF with probability q_t. The values of p_t and q_t are determined by a broadcast command from the central controller, and are in general mutually exclusive, so that either p_t or q_t is always zero. Such a random decision maker could be implemented in many ways. For example, the cell could contain an analog noise source and a comparator that triggers a transition with a small fixed probability p_0. The broadcast command could then be a train of pulses, calculated so that the number of pulses broadcast determines the net probability of transition.

The total number of cells responding to a broadcast command can be thought of as the sum of each decision made by the individual cells, represented as a binary random variable with a probability distribution given by p_t and q_t. Because the cells all share the same probability distribution, the summed number of cells turning on or off will be binomially distributed. Figure 4 shows the shape of the probability distribution, which for large values of N will approach a normal distribution. A broadcast command with probability $(p_t, 0)$ will cause between 0 and $N - N_t^{on}$ cells to turn ON with the binomial distribution

$$P(N_{t+\tau}^{on} = N_t^{on} + x) = \left(\begin{array}{c} N \\ N - N_t^{on} \end{array} \right) p_t^x (1-p_t)^{N-N_t^{on}-x}. \quad (7)$$

Similarly, the probability of x cells turning off given N_t^{on} and a command $(0, q_t)$ is given by the distribution over the number of cells currently ON,

$$P(N_{t+\tau}^{on} = N_t^{on} - x) = \left(\begin{array}{c} N \\ N_t^{on} \end{array} \right) q_t^x (1-q_t)^{N_t^{on}-x}. \quad (8)$$

These distribtions, (7) and (8), constitute a Markov model for the state evolution of the actuator because they depend only on the current state N_t^{on} and the current broadcast command (p_t, q_t).

III. CHOOSING A CONTROL LAW

A full state feedback controller for the cellular actuator takes the form shown in Fig. 5. Given a reference signal y_{ref} and measurements of the current displacement y_t and disturbance force F_d, the controller estimates the number of ON cells using (5) and (6),

$$\hat{N}_t^{on} = (y_t M - F_d N c^{off})/(\eta + c^{on} - c^{off}) \quad (9)$$

An estimate for the desired number of ON cells N_{ref} can be similarly calculated,

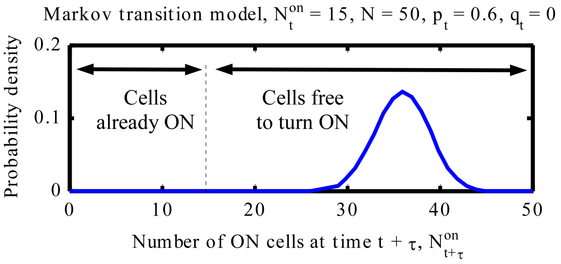

Fig. 4. The aggregate behavior of many cells responding to a broadcast command can be described by a binomial probability distribution. Here a 50 cell actuator is shown, with 15 cells currently on and a broadcast command $p_t = 0.6$, $q_t = 0$.

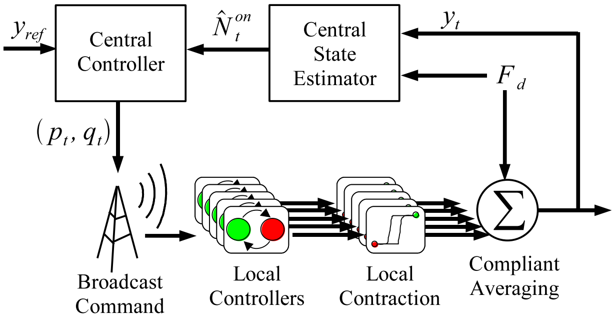

Fig. 5. Closed loop control of the cellular actuator is achieved using a state observer which estimates $\hat{N}_t^{on}$, the number of cells currently contracted, based on measurements of actuator displacement y_t and disturbance force F_d. Using this estimate and the reference position y_{ref}, the controller determines the correct transition probabilities p_t and q_t to broadcast to the cells. The local controllers then respond stochastically to the broadcast signal.

$$N_{ref} = (y_{ref} M - F_d N c^{off})/(\eta + c^{on} - c^{off}) \quad (10)$$

A broadcast command is then calculated using a control law and sent to the local controllers on the cells. Previously, control laws for this artificial muscle architecture were derived using a stochastic Lyapunov function approach [7]. A linear law was chosen of the following form:

$$\begin{array}{ll} N_t^{on} < N_{ref}: & p_t = (N_{ref} - N_t)/N, q_t = 0 \\ N_t^{on} > N_{ref}: & p_t = 0, q_t = (N_t^{on} - N_{ref})/N \end{array} \quad (11)$$

However, convergence in the sense of Lyapunov yields no information about how well the control system can track a reference input. Furthermore, it seems unlikely that this linear law is optimal in any sense. In order to get a quantitiative measure of actuator performance, the dynamic programming framework can be used to pose this control problem as an optimal control problem, where the cost function to be minimized is the response time of the actuator to a step reference. For some background on dynamic programming, chapter 7 of Bertsekas' book on dynamic programming may be useful, covering infinite horizon dynamic programming problems [9].

A. Computing an optimal control law using dynamic programming

The expected time T_c which the actuator takes to converge to a reference N_{ref} from an initial state N_0^{on} can be expressed as an infinite series of local time costs $g(N_t^{on})$,

$$T_c(N_0^{on}) = E\{\sum_{n=1}^{\infty} g(N_{n\tau}^{on})|N_0^{on}\}, \qquad (12)$$

where g is equal to the discrete time interval length τ if N_{ref} has not yet been reached, and zero if N_{ref} has been reached,

$$g(N_t^{on}) = \begin{cases} 0 & , N_t^{on} = N_{ref} \\ \tau, & N_t^{on} \neq N_{ref} \end{cases} \qquad (13)$$

Bellman's equation can be used to write (12) recursively,

$$T_c(N_t^{on}) = g(N_t^{on}) + E\{T_c(N_{t+\tau}^{on})|N_t^{on}, p_t, q_t\} \qquad (14)$$

The control law used to compute p_t and q_t is assumed to be a function of N_t^{on}. This recursive expression for convergence time is convenient because the expected future cost $T_c(N_{t+\tau}^{on})$ can be written as a summation over all possible states using the discrete probability distribution predicted by the Markov model from (7) and (8),

$$T_c(N_t^{on}) = g(N_t^{on}) + \sum_{k=1}^{N} T_c(N_k)P(N_k|N_t^{on}, p_t, q_t) \quad (15)$$

For any convergent control law, (15) can be used to compute the expected convergence time by making an initial guess of $T_c(N^{on})$, and then repeatedly calculating the convergence time using the previous iteration's guess of $T_c(N^{on})$. This technique is called the fixed policy value iteration algorithm. The minimum time to converge $T_c^*(x_0)$ can then be computed by computing the convergence time using iteration, but while minimizing (15) at every iteration,

$$T_c(N_t^{on}) = g(N_t^{on}) + \min_{p_t, q_t} \sum_{k=1}^{N} T_c(N_k)P(N_k|N_t^{on}, p_t, q_t) \qquad (16)$$

B. Value iteration results

The value iteration algorithm was computed for a 50 cell actuator, and used to calculate both the convergence time of the linear control law from (11) and the optimal control law. A plot of the expected convergence time to $N_{ref} = 36$ for 50 cells is shown in Fig. 6. As expected, the optimal control law outperforms the original linear law significantly, particularly when the initial number of ON cells is near the middle of the actuator's working range.

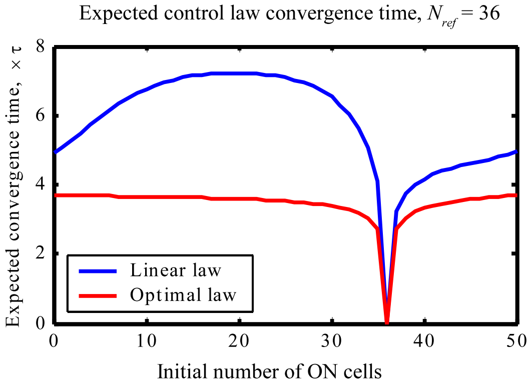

Fig. 6. The expected time as a function of the initial number of recruited cells is shown for several control laws, for a reference $N_{ref} = 36$ cells.

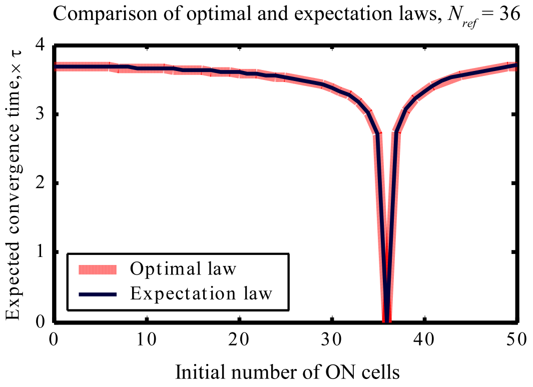

Fig. 7. The law setting the expected value of N_t^{on} equal to N_{ref} has an expected convergence time that lies within less than one percent of the optimal control law for all initial values of N^{on}.

C. An almost-optimal control law

While calculating the optimal control law, it was observed that the values of p_t and q_t produced by the value iteration algorithm were almost exactly equal to the values satisfying the expression

$$E\{N_{t+\tau}^{on}|N_t^{on}, p_t, q_t\} = (1 - p_t - q_t)N_t^{on} + p_t N = N_{ref} \qquad (17)$$

In other words, the control law setting the expected value of $N_{t+\tau}^{on}$ equal to N_{ref} is almost optimal. This is not a surprise. The central limit theorem predicts that the state transition probability distribution should converge to a normal distribution, whose maximally likely value is equal to the expected value. One could think of this law as approximately maximizing the one-step-ahead probability of reaching N_{ref}. To verify the similarity between these two laws, the fixed policy value iteration algorithm was used to compute the expected convergence time of the expectation-based control law:

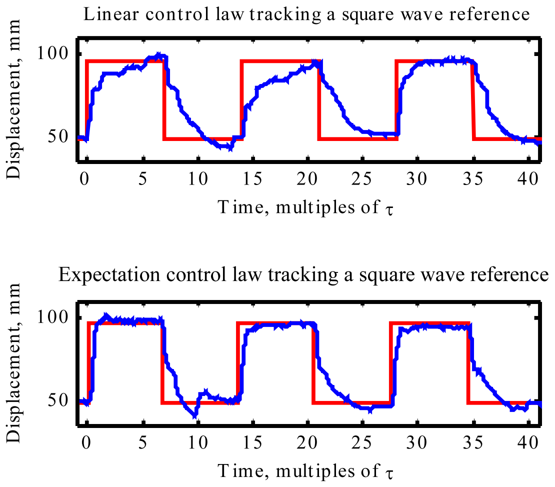

Fig. 8. The expected time as a function of the initial number of recruited cells is shown for several control laws, for a reference N_{ref} = 36 cells. The "stepping" behavior of the actuator, especially visible at the top left, denotes the points at which commands are broadcast.

$$\begin{array}{ll} N_t^{on} < N_{ref}: & p_t = (N_{ref} - N_t)/(N - N_t^{on}), q_t = 0 \\ N_t^{on} > N_{ref}: & p_t = 0, q_t = (N_t^{on} - N_{ref})/N_t^{on} \end{array} \tag{18}$$

The results, shown in Fig. 7, show that the expected convergence time of (18) calculated using value iteration lies within less than a percent of the optimal result.

IV. EXPERIMENTAL VALIDATION

A 50 cell prototype stochastic cellular actuator has been built to evaluate the performance of feedback control laws in the presence of real-world factors such as active material response and sensor noise. The actuator consists of 2 fibers each having 25 cells in series. Helically wound SMA wires (Toki BioMetal, 0.1 mm diameter) 40 mm long are looped eight times through a small piece of perforated circuit board, as shown in the magnified inset of Fig. 9. When turned on, each cell produces roughly 6 mm of displacement at a load of 3.4 N. The actuator is capable of producing a net displacement of 160 mm.

To provide position feedback, the actuator is connected to a tendon that wraps around a capstan connected to a potentiometer. The other end of the tendon is connected to a weight providing a known, adjustable disturbance force. The local control of each cell is accomplished using a current buffer controlled by a computer through a National Instruments I/O card. The local stochastic decision making process and the central controller are implemented in C, using a pseudo-random number generator as a noise source.

To verify that the value iteration calculations shown in Figures 6 and 7 accurately predict the performance of the real actuator, the protype was made to track a square wave having an amplitude equal to a third of the actuator's total range and a period equal to 14 times the sampling interval τ, which was set to 8 seconds. The sampling interval was large in part because the size of the actuator necessitated a very large current source, and the limited amount of current available placed a power constraint on the rate at which the SMA was heated. A smaller, more compact actuator could be made to respond much faster.

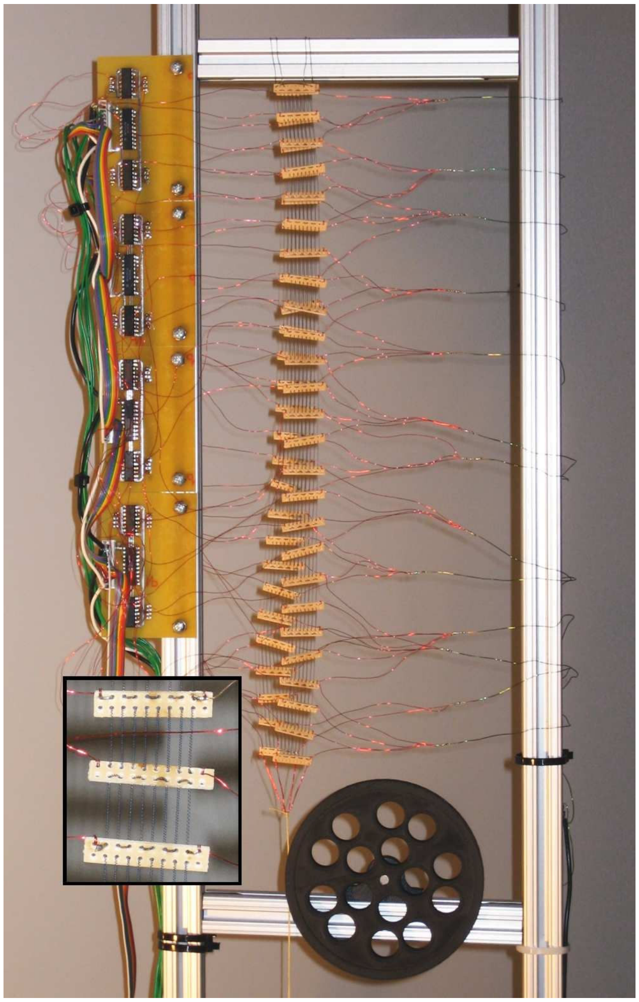

Fig. 9. A prototype actuator having 2 fibers of 25 cells. Two cells are shown in detail at the lower left, made up of helically wound SMA wire. A potentiometer, at the bottom, is used to measure the actuator displacement. Below the potentiometer, a weight is used to provide a known disturbance force to the actuator.

The tracking performance of the two control laws is shown in Fig. 8. As predicted, the linear control law converged to each step change in the reference in 6 or 7 time intervals on average, compared to the 3 or so observed in the expectation control law. A small amount of steady state error was observed in both actuators, but this was presumably caused by limited actuator resolution rather than the control laws.

The slower response of the linear control law may be useful in some circumstances. The expectation control law also exhibited more overshoot than the linear control law, which may be undesirable in some applications. The performance of the expectation control law also degraded more quickly as the sampling interval was shortened, which is probably a symptom of its tendency to overshoot.

V. CONCLUSIONS AND FUTURE WORK

Cellular control architectures are a promising method for controlling artificial muscle materials that are otherwise difficult to control by conventional means. This control technique, based on using local stochastic decision makers within each cell, is a powerful, scalable way to control cellular actuators, because it is based on simple laws of probability whose random uncertainty decreases as more cells are added. This same stochastic nature also makes this control architecture amenable to analysis and optimization using well-established stochastic optimal control techniques such as dynamic programming.

The authors are currently developing a compact cellular actuator designed for robotic applications, having more cells designed for faster response. Various designs for simple, robust local stochastic decision making units are under development, which will enable a departure from the current computer-based testbed. Further work toward better control laws is also progressing. The state estimator presented in this paper makes steady-state assumptions that limit the control system performance when the sampling interval τ is decreased. An estimator relying on a brief time history of the actuator's position and commands could dramatically improve the accuracy of state estimation, even in the presence of significant transient behavior in the active material.

REFERENCES

[1] K. Cho, H. Asada, "Architecture Design of a Multi-Axis Cellular Actuator Array Using Segmented Binary Control of Shape Memory Alloy," in *IEEE Transactions on Robotics*, v.22, n.4, pp.831-842 August 2006.

[2] J.D.W. Madden et. Al, "Artificial muscle technology: physical principles and naval prospects," *IEEE J. Oceanic Engineering*, v.29, n.3, July 2004, pp. 706-728

[3] Y. Bar-Cohen et. Al, *Electroactive Polymer Actuators as Artificial Muscles*, SPIE Press, 2001

[4] D. De Rossi et. Al, "Compliance control and Feldman's muscle model," *IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2006

[5] B. Selden, K. Cho, H. Asada, "Segmented binary control of shape memory alloy actuators using the peltier effect," *Proc. IEEE ICRA*, 2004, pp. 4931-4936

[6] J. Ueda, L. Odhner, H. Asada, "A broadcast-probability approach to the control of vast DOF cellular actuators," *Proc. IEEE ICRA*, 2006, pp. 1456-1461

[7] J. Ueda, L. Odhner, S. Kim, H. Asada, "Distributed stochastic control of MEMS-PZT cellular actuators with broadcast feedback," *IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2006

[8] S. Fox, *Human Physiology, Seventh Edition*, McGraw-Hill, 2002

[9] D. Bertsekas, *Dynamic Programming and Optimal Control*, Athena Scientific, 2005