2005 Conference on Information Sciences and Systems, The Johns Hopkins University, March 16–18, 2005

Power Control for Target Tracking in Sensor Networks

Jean-Paul Wagner
EPFL
Lausanne, CH-1015, Switzerland
Jean-Paul.Wagner@epfl.ch

Răzvan Cristescu
Caltech
136-93 Pasadena, CA 91125, USA,
razvanc@caltech.edu

Abstract — **We consider the problem of a sensor network tracking a moving target that exhibits a Markov model of mobility. The sensor nodes have adjustable power levels and the precision of the measurement of the target location depends on both the relative distance from the target to the measuring sensor, and on the sensing power level used by that sensor. An important issue in sensor networks is the power efficiency, thus we consider the optimization of a family of cost functions that include both the accuracy of a measurement and the power used to do that measurement. We define our problem as a control policy optimization for a partially observed Markov chain. For such scenarios, we derive optimal power control policies based only on partial observations of the target location, and propose hand-off techniques based on this policies.**

I. Introduction

Target tracking and power control

Consider a sensor network that tracks a moving target. We assume that each sensor[1] has an adjustable power level for sensing, and the accuracy of the target position measurement, performed by the measuring sensor, is determined by both the relative distance from the target to the sensor, and by the power used by the respective node for sensing.

As battery power is the scarce resource in sensor networks, an important goal is to minimize the sensor power consumption, while maintaining a good measurement accuracy. This task can be done by an adequate power adjustment mechanism implemented by the sensor, and dependent on the target mobility model. The power control policy should accommodate several concurrent effects, namely, in most scenarios (a) for a fixed power level, the accuracy of a measurement decreases with the increase of the distance, and (b) for a fixed desired accuracy, the necessary power level increases with the increase of the distance. Thus, for the design of power control policies, we will consider optimizations of cost functions that include both these effects, in terms of measurement accuracy and power consumption. Also, in practical scenarios, the power control policy has to rely only on limited information available locally at the sensor node about the state of the target, as a result of previous measurements. Moreover, in order to minimize the battery consumption, sensors should limit communication with the neighboring sensors.

An important task in this scenario is the *hand-off* operation. In a typical sensor network, sensor nodes can communicate with other nodes in their transmission range. We consider a proximity relevance scenario, where the importance of having an accurate localization of the target increases when the target is closer to the sensor. For instance, this scenario has practical importance when sensors represent guarded sites, and targets represent unwanted intruders. Hand-off is defined as the decision taken by a sensor tracking the target to delegate the tracking task to another sensor. This decision is usually taken if the target becomes closer to another sensor than to the sensor that currently does the measurement. Then the tracking is handed-off from the former sensor to the latter. Note that this is not a trivial task, since (a) the sensor currently doing the measurement has only *partial* information about the target position, and (b) the other sensors are in an idle, low-power state, when they are not assigned the sensing task.

The goal of this work is to derive a power control policy based only on partial observations locally available at the measuring sensors about the state of the target, by considering jointly the power consumption and accuracy of measurement. We also propose a decentralized hand-off technique based on this optimization. To the best of our knowledge, the novelty of our work stands in jointly addressing control and sensing in tracking targets by sensor networks, under uncertainty conditions.

Related work

Target tracking in sensor networks is addressed in [2], [4], [5]. Our model for the target tracking system is inspired by the elegant framework provided in [6]. In that work, the authors use the notion of belief state as a means of quantifying the usefulness of communication among sensors for tracking a mobile target under location incertitudes. In our model, we also use this belief state (also called information state) to derive an optimal control power policy for a sensor tracking a Markovian moving source; this control policy influences in turn the information state. On the lines of [6], we also provide criteria for soft hand-off in a sensor network implementing our proposed power control.

Main contributions and organization of the paper

We define a system model for tracking a Markov target by relying only on partial observations. We consider the optimization of a family of cost functions that include both the accuracy of the observations and the power level used. For various relevant costs, we derive optimal policies for the power adjustment control, based on the theory of Markov chains under partial observations. Finally, we propose hand-off techniques based on such control policies.

In Section II we present a one-dimensional system model of tracking a target under partial observations. We introduce a family of cost functions that consider both the accuracy of measurement and the power consumption, and derive an optimal control policy to optimize these cost functions. In particular, we consider the case of a normally distributed probability

[1] We use *sensor*, *node*, *sensor node* interchangeably along this paper.

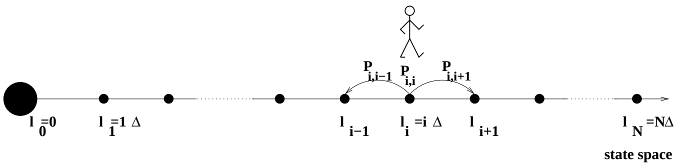

Fig. 1: One-dimensional moving target with a Markov behavior.

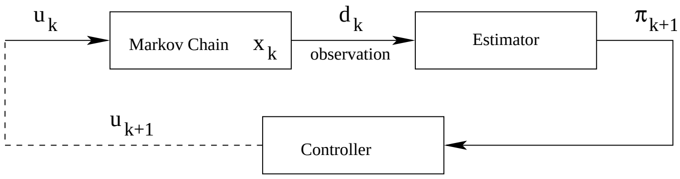

Fig. 2: The one-dimensional control system.

to model the accuracy of the location observation, and we present a fast approximated algorithm for the optimal control policy. In Section III we use our insights from the one-dimensional case to derive an approximated policy for target tracking on a two-dimensional grid. Further, in Section IV, we propose hand-off techniques based on these control policies, and we analyze these algorithms by numerical simulations in Section V. We conclude with Section VI.

II. System Model

Consider that the target moves following a discrete-time Markov chain behavior. This means that the position x_k of the target (the *state*) at time k depends only on its position x_{k-1} at time $k-1$, the successive positions of the target thus forming a Markov chain. We will begin by considering the simplified case of a single sensor tracking a target that moves in a one-dimensional space (on a line). In Section III we generalize this setting to a two-dimensional state space.

Without loss of generality, we consider that the positions of the target are constrained to the uniform grid on the line, that is the set of positions $\mathcal{L} = \{l_0 = 0, l_1 = \Delta, \ldots, l_i = i\Delta, \ldots, l_N = N\Delta\}$, with Δ a positive real number. The measuring sensor is placed in position $l_0 = 0$ (see Figure 1). Note that this setting can be generalized in a straightforward manner to the case when the sensor is placed at an arbitrary position l_i on the line, or to the case of a non-uniform grid.

One-dimensional model

System dynamics

Consider the decisions taken by a sensor on the power level used at a certain time moment k. The sensor takes a decision u_k, which is modelled as the power level used by the sensor at time k. By using the power level u_k for the measurement, the sensor observes a position d_k of the target. In practical situations, the *observed* distance d_k is not the actual state $x_k \in \mathcal{L}$ of the target, but rather a random variable with probability distribution centered on the state x_k, that models the uncertainty of estimation: the closer d_k is to x_k and the larger the sensing power u_k, the better the estimation. In other words, when observing the state x_k, the sensor can only get a *partial* observation d_k about the state. Using this partial observation d_k, as well as all the previous observations[2] d^{k-1} and all the previous controls u^{k-1}, the sensor can make an estimation on where the target will most probably be at time $k+1$. Note that the evolution of the actual state of the target is independent of the control applied at the sensor (see Figure 2).

The goal is to find an optimal policy of control decisions u_k that the sensor has to take, in order to estimate (on average) the state as precisely as possible, given a constraint on the consumed energy.

Without loss of generality, assume the power level takes values in $[0, u\mathrm{max}]$. To summarize, our system is described at time k by the following parameters:

- $x_k \in \mathcal{L}$ – actual position (but unknown) of the target at time k: the *hidden* chain state.
- $u_k \in [0, u\mathrm{max}]$ – controllable power level of the sensor: the *control* applied to the chain at moment k.
- $d_k \in \mathcal{L}$ – random variable with probability distribution $p(d_k|u_k, x_k)$: the partial *observation* about the actual state of the chain.

Also, the following information is assumed a-priori known:

- The transition probability matrix P of the hidden Markov chain x_k: $P_{i,j} = p(x_k = i|x_{k-1} = j)$ (these transitions are assumed independent of the sensor instantaneous power u_k, as in most applications the mere act of sensing does not influence the sensed data). For instance, in a practical setting, P contains information about the terrain in which the target evolves: an up-hill slope in a given direction can be modelled as a lower transition probability in that direction.
- The initial probability density vector over the states $[p(x_0 = l_i)]_{i=1}^N$ (in practice, this can be initialized as the uniform probability distribution over the states).
- The probability of occurrence of an observation d (probability vector), when the target is in state x and the sensor control is u: $p(d|x, u)$, which is a distribution with mass centered on the real state of the target. As the power level u increases and the distance between the sensor and the real (but unknown) state decreases, this distribution has typically its mass more concentrated around the real state x (see Figure 3).

Based on these assumptions, we can express the dynamics of the system as:

$$\begin{aligned} x_{k+1} &= f(x_k) \\ d_k &= g(x_k, u_k), \end{aligned}$$

for $k \geq 0$. Here, f and g are the state transition and respectively observation functions of the hidden Markov chain, as defined above in this section (see Figure 2).

We will define later in this section meaningful cost functions $c(x, u)$ of the control and state. For a given cost function, the optimal control for such systems can be fully characterized by a quantity called the information state [3]. A good choice for this quantity is the conditional probability π_k of the state given all past controls and observations, $[\pi_k(i)]_{i=0}^N = [p(x_k = l_i|u^{k-1}, d^{k-1})]_{i=0}^N$. This information state defines the probability of the target being in state l_i at time k, given the sequence of controls u^{k-1} and observations d^{k-1}.

Moreover, we derive a recursion formula, implemented by the estimator box in Figure 2, that allows updating the information state π_{k+1} by using only π_k and the newly acquired u_k, d_k:

$$\pi_{k+1} = \frac{1}{C} \pi_k D(u_k, d_k) P \qquad (1)$$

[2]For any sequence z_k, we will denote $z^{k-1} = \{z_0, z_1, \ldots, z_{k-1}\}$.

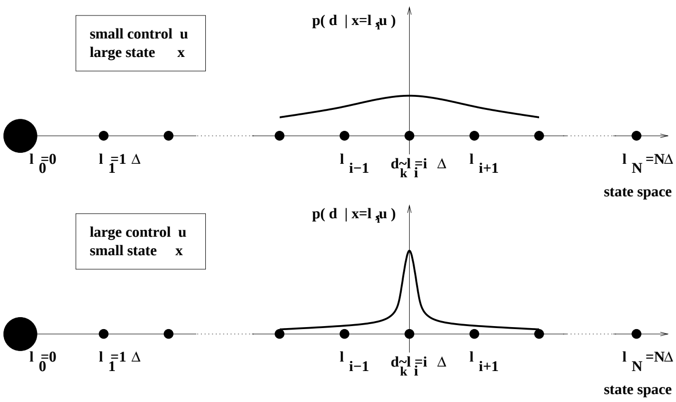

Fig. 3: An observation is more relevant if the entropy of $p(d|x,u)$ is small.

where C is a normalization factor which ensures that the result is a probability vector, and $D(u_k, d_k) = \text{diag}(\{p(d_k|x_k = l_i, u_k)\}_{i=0}^{N})$ is a diagonal matrix. For the sake of space we omit the derivation of this update formula.

We describe now a family of cost functions which include the meaningful constraints of our model: the information about the state of the target and the level of power consumed by the sensor.

Cost function

The choice of the cost function is application dependent. In this section we will particularize our model to a set of specific practically motivated parameters, however the same methodology for finding optimal control policies remains valid if various cost functions are considered.

In our model, we assume the observations a sensor gets about the target position are random, and given by normal probability densities, centered on the actual state x_k, and with variance dependent on the power used by the sensor and on the distance between the target and the sensor. Namely, the variance of the probability density function of d_k decreases with the increase of u_k and when the target is close to the sensor (see Figure 3). We thus consider a tradeoff[3]: our goal is to reduce the uncertainty of the observation, and thus the entropy $p(d_k|x_k, u_k)$, while using a small power level u_k:

$$\varphi(u,x) = C_1 F_1(H\left(p(d|x,u)\right)) + C_2 F_2(u) \tag{2}$$

The first term of this sum, $F_1(\cdot)$, is a function of the entropy of the observation, and the second term $F_2(\cdot)$ is a function of the power used to make this observation, where both $F_1(\cdot), F_2(\cdot)$ are monotonically increasing functions, and C_1, C_2 are weighting constants. Note that the cost function (2) is similar in form to that used in [3], however in our cost function we consider the dependence of both optimization terms on the sensor power level, which in our model is a *controllable* quantity.

The cost function definition (2) is valid for any probability distribution of the observation; for the sake of simplicity, we will limit our further discussion to normally distributed random variables. For a discrete normal random variable, the

[3]Note that minimization of only the entropy implies continuously using high power levels, which might be highly inefficient in a practical scenario.

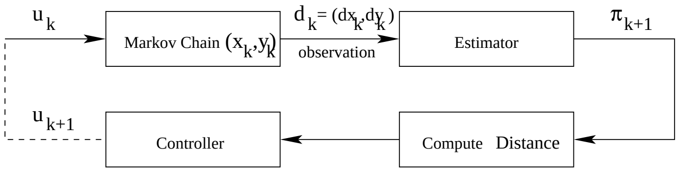

Fig. 4: The two-dimensional control system with a two-dimensional Markov chain.

entropy term in (2) can be written as [1]:

$$H\left(p(d|x,u)\right) \approx \frac{1}{2}\log(2\pi e \cdot \sigma^2(x,u)) - \log\Delta, \tag{3}$$

where the approximation tends to be exact as the quantization step Δ of the state space tends to zero; we make the realistic assumption that the variance $\sigma^2(x,u)$ of the distribution is directly proportional to the distance between the sensor and the estimated state, and inversely proportional to the used power:

$$\sigma^2(x,u) = \alpha\frac{h_1(x)}{h_2(u)} \tag{4}$$

where $h_1(\cdot), h_2(\cdot)$ are also monotonically increasing functions, and α a weighting constant.

Optimal control

The goal is to find a policy u^k that minimizes $\sum_{k=1}^{\infty}\varphi(u_k, x_k)$. The optimal control (implemented by the controller box in Figure 2) depends only on the information state [3], however the optimization is rather complicated in practice, since the information state space is defined over the space of all probability vectors. Instead, we will use the information state to compute an estimate $\bar{x}_{k+1}$ of the target state at time $k+1$, and use this *estimated* state (instead of the information state) to determine an approximate controller. Given the instantaneous π_k, an estimation of the current state can be efficiently computed:

$$\bar{x}_k = \sum_{i=0}^{N} l_i \pi_k(i) \tag{5}$$

Using $\bar{x}_k$ as estimated state, we approximate the optimal controller by minimizing the instantaneous cost (2):

$$u_k = \arg\min_u \varphi(u, \bar{x}_k). \tag{6}$$

Note that this approximation is sub-optimal, however it provides a simple and fast means for the computation of an intuitively good approximation of the optimal controller.

III. Two-Dimensional Model

We consider now a Markovian target moving in a two-dimensional state space (x,y), namely the position (x_k, y_k) of the target at time k depends only on its position (x_{k-1}, y_{k-1}) at time $k-1$. Thus, the development in the previous section can be generalized in a straightforward manner to the two-dimensional case. Analogously to the previous section, we can compute an estimated distance from the estimated position vector and use it to compute an optimal control for the two-dimensional Markov chain (see Figure 4).

However, a practical implementation of this system is computationally difficult. The states now form an $N \times N$ two-dimensional array, which in the one-dimensional formulation

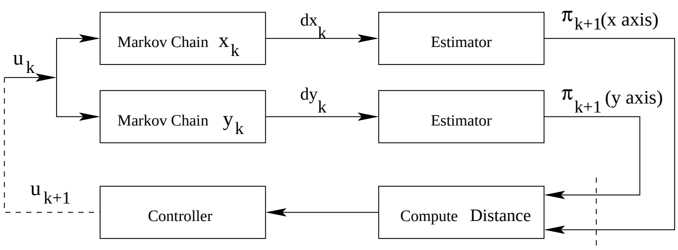

Fig. 5: The two-dimensional control system with two one-dimensional Markov chains.

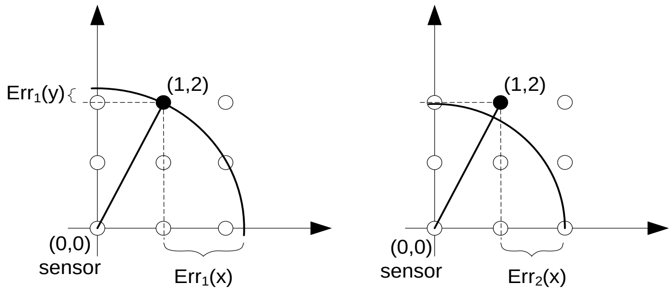

Fig. 6: (*left:*) $Err_1(x)$ and $Err_1(y)$ are the errors made when we use the distance between the sensor (0,0) and the estimated state (1,2). (*right:*) $Err_2(x)$ is the error made when we use $\max(x, y)$: at least one of the two Markov Chains has a well fit controller.

is equivalent to a vector of length N^2. Moreover, expressing the transition matrix P is no longer trivial since certain states can be close neighbors in the two-dimensional state space, but no longer in the expanded one-dimensional version. Instead, we consider yet another approximation, namely our approach is separate the two dimensions of the problem into disjoint Markov chains. As we will see later by numerical simulations, for loosely coupled Markov transitions on the two dimensions, this approximation yields to good results, while being computationally simple.

Thus, we consider the target movement of being composed of two independent Markov chains, each governing the movement of the target along one dimension. Each coordinate is estimated at time $k+1$ from the decisions u^k, the observations dx^k and dy^k on the axes x and y respectively, and the estimated one-dimensional states at time k. We then use these coordinates to compute an estimated state distance, which in turn is used to compute the next control (see Figure 5). Finally, the resulting control is applied to both Markov chains. It is clear that in most cases this control will not be optimal for any of the two chains. Figure 6 (left) shows the nature of the error made in the state space. The distance on which the power decision are based is always larger than any of the coordinates, which means that the sensor will always use more power than the optimum. In other words, the entropy term in the cost function is favored, on the expense of the used power. One intuitively better alternative is shown in Figure 6 (right): the estimation error can be reduced by rather computing the control as a function of $\max(x, y)$, instead of both estimated coordinates. This procedure ensures that at least one of the two chains has a well fit controller. In any case, this optimization only refers to the separated model: in reality the target moves in a two-dimensional space and at a distance from the sensor that is best approximated by the two-dimensional euclidean distance, not by $\max(x, y)$.

To summarize, our algorithm for tracking a two-dimensional moving target is as follows:

1. Get an observation dx_k and an observation dy_k.
2. Estimate the coordinates x_{k+1} and y_{k+1} using the newly acquired observations and all the previous information (contained in the information state vector π_k). Until this point the two coordinates are estimated in parallel and independently.
3. Based on these two new coordinate estimates, compute the distance of the estimated state or compute the maximum coordinate.
4. Feed the resulting value in the controller, which yields the control u_{k+1} that minimizes the cost function for the given input.
5. Apply this control u_k to observe both Markov chains, and repeat.

IV. Hand-Off Algorithms

We suppose that there is always just one sensor tracking the target using the model we described earlier. At each time k there is thus one active *tracking sensor*. The other sensors in the network are *idle sensors*. The aim of the hand-off algorithm is to determine the tracking sensor so that the power consumption is minimized and the precision is maximized, i.e. the hand-off algorithm needs to find the sensor for which the cost function is minimized, among all sensors in the network.

Based on the model we used for the two-dimensional case, we propose a simple (partly randomized) algorithm for hand-off in a network of target-tracking sensors. The algorithm we present is sub-optimal and involves several approximations, but our numerical simulations in Section V show that it yields good results.

Partly randomized hand-off algorithm

- At every time interval T each idle sensor performs a sensing operation using a small fixed amount of the available power.
- Each sensor gets a rough estimate of the two coordinates which are drawn from Gaussian distributions, with variance depending on the roughly observed state, and on the used low power level.
- The roughly observed state distances at every idle sensor are compared to the distance of the current state estimate at the tracking sensor (which, as opposed to all the idle sensors, used the Markovianity of the system to refine the observation).
- If one of the idle sensors reports a smaller roughly observed distance than the one the tracking sensor has currently estimated, and if this rough observation was made using less power than the optimal power level determined at the tracking sensor, then this sensor is probably closer to the estimated state and thus it can get an observation using less power.
- The sensor selected by this criterion will be the tracking sensor for the next period T when the hand-off algorithm is run anew.

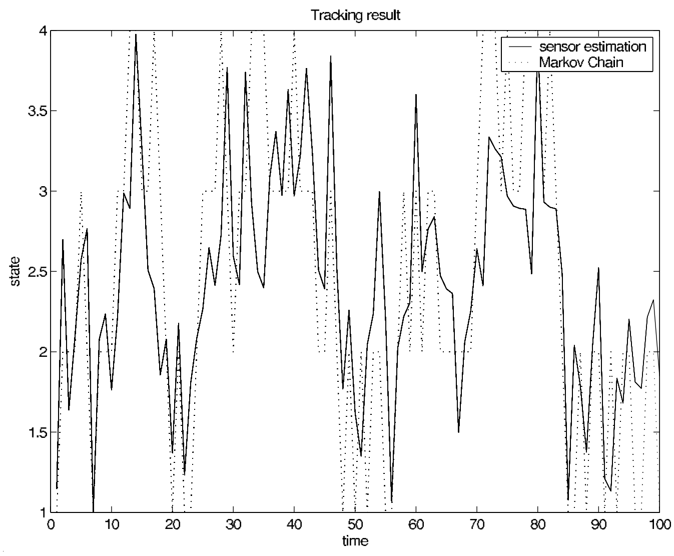

Fig. 7: One-dimensional system: tracking result with $p = 0.33$.

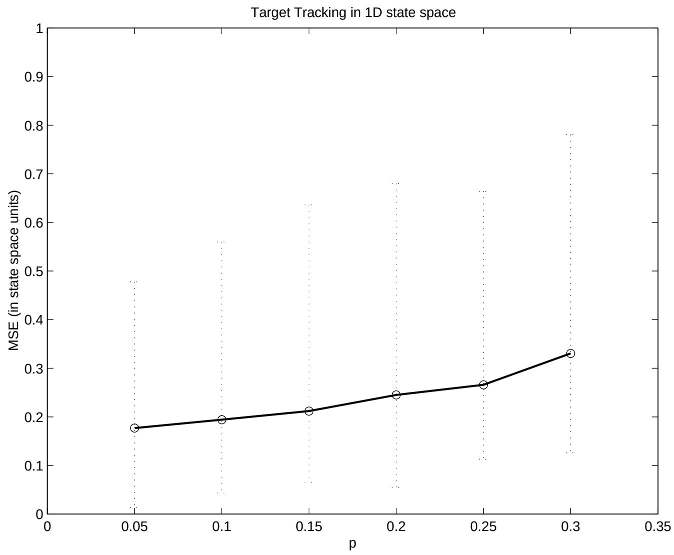

Fig. 8: One-dimensional system: MSE for different birth-and-death probabilities.

Soft hand-off criterion

The above outlined algorithm demands that all of the sensors in the network have some means to communicate among themselves, as a hand-off between *any* two sensors is possible. A more realistic assumption would be that any sensor can at least communicate with some of its neighboring nodes, and thus only hand-offs between neighboring nodes can be achieved in the network (*soft* hand-offs).

We expect that the above algorithm can achieve soft hand-offs if the scheduling interval (wake-up rate of the idle sensors) is appropriately chosen. We also expect that this interval T should depend on the structure of the Markov chain's transition probability matrix P, which in turn depends on the speed and direction of the target, and on the sensor density. The investigation of such a soft hand-off criterion is a subject of our current research.

V. Numerical Simulations

One-dimensional model

This subsection presents the results using the system outlined in Section II. Figure 7 shows the estimated target evolution, as compared to the actual evolution to which the sensor has no access, for a target moving in a $N = 4$-state space. The target can thus take 4 equidistant discrete positions on a line. The movement is governed by a birth-and-death Markov process with $p = 0.33$ and there is a single tracking sensor at the origin of the state space.

The cost function we chose for the simulations in this section is given by:

$$\varphi_{sim}(u_k, \bar{x}_k) = \frac{1}{2}\log_2\left(\frac{\bar{x}_k}{u}\right)^2 + \frac{u^5}{\bar{x}_k^3} \qquad (7)$$

This cost function was set empirically. It is derived from (2), (3) and (4) by setting $\sigma = \frac{\bar{x}_k}{u}$ and $F_2(\cdot) = \frac{u^5}{\bar{x}_k^3}$. These settings are to be considered as representatives of a family of valid choices, which depend largely on the characteristics of the modelled sensor network.

The range of the sensor is modeled to be equal to 3-state space units: if the estimated distance is equal or greater than 3, the sensor will use maximum power, and can not improve the precision of the observation any further.

Under these settings, Figure 8 shows the MSE, in state space units, between the estimated position and the actual target position. We show the results for different probabilities of birth-and-death probability p. Note that, in practice, the higher p can be interpreted as higher target speed. For each p, the values shown are averages over 100 realizations of the 4-state Markov Chain, each realization being over 100 state transitions. The tracking sensor is always located at the origin of the state space.

It can be seen that the tracking results are rather robust, with a slight decrease in precision as the target changes states faster.

Two-dimensional model with handoff

This subsection presents the results for a two-dimensional tracking system, as outlined in Section III. The settings for these simulations are similar to those outlined in the above subsection with the following differences and additions:

- The movement along each coordinate is governed by a Markov chain with $N = 16$ equidistant states.
- For each realization, the probability of birth-and-death p is the same for both of the independent Markov chains.
- 10 sensors are uniformly distributed in the 16x16 state space. The tracking sensor is determined using the handoff algorithm presented in Section V. The algorithm is run every $T = 10$ time units.

Under these settings, Figure 9 shows the MSE, in state space units, between the estimated position in the two-dimensional space and the actual (hidden) target position. For each p, the values shown are averages over 100 realizations of the two 16-state Markov chains, each realization being over 100 state transitions.

Finally, Figure 10 shows the controls taken during one of the realizations used to get the above averages ($p = 0.33$). The

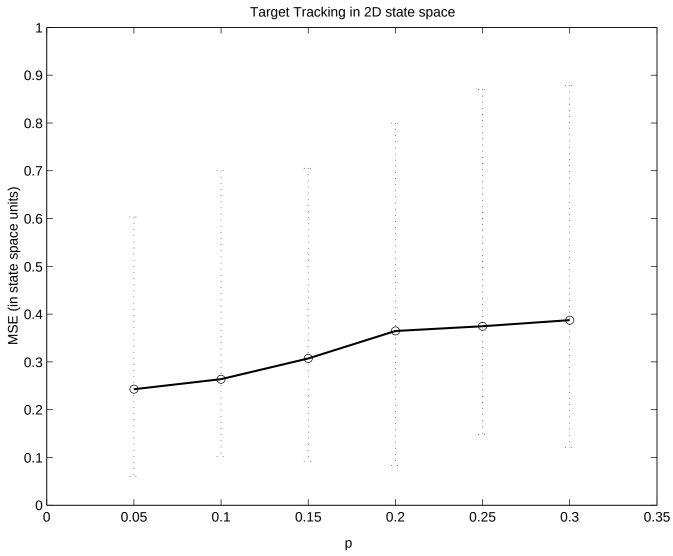

Fig. 9: Two-dimensional model: MSE for different birth-and-death probabilities.

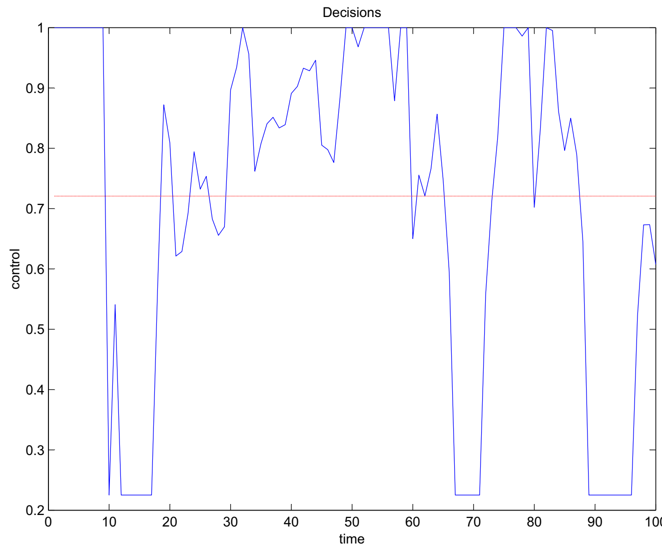

Fig. 10: Two-dimensional model: power usage in the network. The deep decreases in consumption at times 10 and 90 for example are attributed to the hand-off algorithm: a sensor that uses a high amount of power passes the sensing task to a sensor that is expected to be closer to the target and which in turn makes higher precision observations using less power.

values shown are the normalized power levels as decided by the controller. At each time, only the tracking sensor contributes to the power consumption attributed to the sensing task. Note that the average power consumption in the network is equal to a single sensor using 70 to 75% of its power continuously.

VI. Conclusions and Future Work

In this paper we have addressed the problem of tracking a moving target using a sensor network, both in a one and two dimensional state space. We have set up our model under realistic assumptions, from which we derived an optimal control policy, in terms of power usage and tracking reliability, that can be implemented efficiently at the sensor level. The model is robustly validated using our numerical simulations.

Our current research efforts are focused on the analysis of stability and robustness of the control policies corresponding to the family of cost functions we proposed. An important subject of further investigation is the study of the interdependence among system parameters such as sensor density, target speed and hand-off interval, and their influence on the optimal control policies.

References

[1] T. M. Cover, J. A. Thomas, *Elements of Information Theory*, Wiley, 1991.

[2] A. Javed, Z. Butler, F. Constantin, V. Crespi, G. Cybenko, D. Rus, *Tracking a Moving Object with a Binary Sensor Network*, ACM Sensys 2003.

[3] P. R. Kumar, P. Varaiya, P., *Stochastic Systems: Estimation, Identification and Adaptive Control*, Prentice Hall, 1986.

[4] S. Pattem, S. Poduri, B. Krishnamachari, *Energy-Quality Tradeoffs for Target Tracking in Wireless Sensor Networks*, IPSN 2003.

[5] F. Zhao, J. Liu, J. Liu, L. Guibas, J. Reich, *Collaborative Signal and Information Processing: An Information Directed Approach.*, Proceedings of the IEEE, 2003.

[6] F. Zhao, J. Shin, J. Reich, *Information-Driven Dynamic Sensor Collaboration for Tracking Applications*, IEEE Signal Processing Magazine, March 2002.