Available online at www.sciencedirect.com

ScienceDirect

Digital Signal Processing 19 (2009) 978–989

Digital Signal Processing

www.elsevier.com/locate/dsp

Approximate stochastic dynamic programming for sensor scheduling to track multiple targets

Y. Li [a,*], L.W. Krakow [b], E.K.P. Chong [a,b], K.N. Groom [c]

[a] *Department of Electronic and Computer Engineering, Colorado State University, Fort Collins, CO 80523-1373, USA*
[b] *Department of Mathematics, Colorado State University, Fort Collins, CO 80523-1874, USA*
[c] *Sandia National Laboratories, Albuquerque, NM 87185, USA*

Available online 25 May 2007

Abstract

The problem of sensor scheduling is to select the number and combination of sensors to activate over time. The goal is usually to trade off tracking performance and sensor usage. We formulate a version of this problem involving multiple targets as a partially observable Markov decision process, and use this formulation to develop a nonmyopic sensor-scheduling scheme. Our scheme integrates sequential multisensor joint probabilistic data association and particle filtering for belief-state estimation, and use a simulation-based Q-value approximation method called completely observable rollout for decision making. We illustrate the effectiveness of our approach by an example with multiple sensors activated simultaneously to track multiple targets. We also explore the trade-off between tracking error and sensor cost using our nonmyopic scheme.

Keywords: Sensor scheduling; Stochastic dynamic programming; Multiple target tracking; Partially observable Markov decision process; Particle filtering

1. Introduction

The purpose of sensor scheduling is to select the number and combination of sensors to activate over time. A typical goal is to trade off tracking performance and sensor usage. Sensor-scheduling problems are addressed in [1–3], where they are formulated as optimization problems to minimize the instantaneous estimation error or to maximize the information gain. Because these schemes only consider the instantaneous performance, they are said to be "myopic." Nonmyopic sensor scheduling has gained interest, focusing on the use of stochastic dynamic programming, for example in [4–6].

We formulate the sensor-scheduling problem as a partially observable Markov decision process (POMDP) to include long-term performance considerations [4,5]. The underlying process in the POMDP framework [7,8] is a controlled Markov process. The sensor-scheduling decision in a POMDP is based on recursively calculating the *belief state*, the posterior distribution of the underlying state given the history of measurements and sensor-scheduling

* Corresponding author.
E-mail addresses: liyun@engr.colostate.edu (Y. Li), lwkrakow@lamar.colostate.edu (L.W. Krakow), echong@engr.colostate.edu (E.K.P. Chong), kngroom@sandia.gov (K.N. Groom).

1051-2004/$ – see front matter
doi:10.1016/j.dsp.2007.05.004

actions. In general, analytical calculation of the belief state is impossible. In this paper, we employ a Monte Carlo approach that combines two techniques: particle filtering for belief-state estimation, and a simulation-based Q-value approximation method for decision making via "lookahead." Particle filtering [9,10] is a Monte Carlo method for belief-state estimation. At each time step, the output of the particle filter is a set of particles (samples) that represents the current belief state. The simulation-based Q-value approximation method uses Monte Carlo simulation to evaluate, at the current belief state, the expected cumulative cost of candidate actions as approximations of Q-values, which are then used to select the optimal action. As particle filtering provides a set of particles for the Q-value approximation method to initiate the evaluation, these two techniques dovetail naturally in our approach.

Because of the POMDP formulation, our approach can take both long-term and short-term costs into consideration. Furthermore, because it is a Monte Carlo method, it does not rely on analytical tractability, and therefore it can incorporate sophisticated target dynamics and sensor models.

In our previous work [11–13], particle filtering and policy rollout (a simulation-based Q-value approximation method) were applied to the scenario with single sensor activation for single-target tracking. The work presented here is an extension of our previous work to the multisensor multitarget case. (A preliminary version of this work was presented in [14].) A number of new issues are solved in this extension. In the belief-state estimation, there is the *data association* problem to decide which target is associated with each observation, and the *sensor data fusion* problem to combine information from multiple sensors. We develop an innovative tracking algorithm that integrates multisensor data fusion, multitarget data association, and particle-filter tracking techniques. For decision making, we develop a variation of policy rollout called completely observable rollout (CO-rollout). In this paper, the problem formulation and the sensor model are different from those of [11–13]. We use the algorithm here for studying the trade-off between tracking error and the sensor usage cost.

Our experiments involve the activation of up to two of four sensors to track two targets. The results verify that our integrated multisensor joint probabilistic data association (MS-JPDA) and particle-filter tracking algorithm works correctly, illustrate that our Q-value approximation method is effective in improving the total cost in heterogeneous sensor networks, and show the trade-off between tracking performance and sensor usage cost.

2. Problem formulation

A POMDP [7,8] is specified by its state space $\mathcal{X}$, action space $\mathcal{U}$, observation space $\mathcal{Y}$, state transition law $K(X'|X,u)$ ($X, X' \in \mathcal{X}$, and $u \in \mathcal{U}$), observation law $L(Z|X,u)$ ($Z \in \mathcal{Y}$), initial state distribution p^0, and one-step cost function $r(X,u)$. It is basically a Markov decision process (MDP) where the state is only partially observable through L.

Starting at time 0 from the initial state X^0 with known distribution p^0, a POMDP evolves as follows. At time step k, the state of the system is X^k and the observation Z^k is available. Then the action u^k is selected and a cost $r(X^k, u^k)$ is incurred. After that the system moves to the state X^{k+1} according to the transition law $K(X^{k+1}|X^k, u^k)$, and an observation Z^{k+1} is generated randomly according to the observation law $L(Z^{k+1}|X^{k+1}, u^k)$.

Since the state is not directly observable, a POMDP keeps track of the belief state b^k, defined as $p(X^k|I^k)$, the posterior probability distribution of state X^k conditioned on the observation and action history $I^k := (p^0, u^0, Z^1, u^1, Z^2, \ldots, u^{k-1}, Z^k)$. The goal here is to choose an action u^k, based on the belief state b^k, from a set of available actions $U(b^k)$ to minimize the expected total cost. A *policy* is defined as a sequence of mappings from belief states to actions $\pi = \{\pi^k\}$.

Let the expected total cost starting from initial belief state p^0 and using policy π over a horizon of H steps to be $J_H(p^0, \pi) = E(\sum_{k=0}^{H-1} r(X^k, u^k) \mid p^0, \pi)$, where $u^k = \pi^k(b^k)$, and the expectation is taken over all possible state and observation sequences. The objective here is to find an optimal policy $\pi^* = \{\pi^{*k}\}$ to minimize $J_H(p^0, \pi)$.

We define the cost at belief state b^k by taking action u as $g(b^k, u) = \int r(X,u) b^k(X)\, dX$. Then the Q-value of action u at belief state b^k is

$$Q_{H-k}(b^k, u) = g(b^k, u) + E\big(J^*_{H-k-1}(b^{k+1}) | b^k, u\big), \tag{1}$$

where $J^*_{H-k-1}(b^{k+1})$ is the optimal value over $H-k-1$ time steps starting at the next belief state b^{k+1}. Bellman's optimality principle for POMDPs states that the minimum expected total cost is given by $J_H(p^0, \pi^*) = \min_u Q_H(p^0, u)$, and the policy that selects action $\pi^{*k}(b^k) = \arg\min_u (Q_{H-k}(b^k, u))$ at step k is optimal. Because the Q-value of an