5.3 Finite State Space Problems: POMDPs

[2]The filtering problem for hidden Markov models (HMMs), which is essentially the problem of evaluating the cost of a fixed control policy under imperfect information, is solvable analytically in two main cases. The first case is the linear Gaussian model, presented in section 5.2, where the optimal filtering problem is solved by the Kalman filter. The second case concerns models with a finite state space X. In both cases, we have an efficient way of computing the conditional distribution $P_{x_k|\mathcal{I}_k}$ of section 5.1 recursively following the Bayes filter. In the linear Gaussian case, This turned out to be the most difficult problem to solve, since the optimal control law was then simply the same as in the perfect information case, replacing the true state by the conditional mean estimate. In the general finite-state space case things are not quite so simple however. First, we will assume that the observation and control spaces are also finite. Then to compute the optimum controller *we must still work with the full conditional distribution* (a function mapping the observations into then $n-1$ dimensional simplex if X has n states) and there is no separation principle, which leads to a significant increase in computational complexity. In fact, the optimal control problems in this framework are in general intractable in a precise sense (typically PSPACE-complete)[3]. Seminal work in this area can be found in the control and operations research literature of the 1960's and 1970's [Ast65, SS73] where the term POMDP (Partially Observed Markov Decision Process) was coined. There is nothing in this terminology that indicates a finite state space, but this is typically was is meant when it is used. In this section we outline a few basic facts of the theory of POMDPs, in particular the *concave piecewise linear form of the finite-horizon value function*. Most recent work focuses on approximate methods, and there are also still some basic theoretical questions open concerning the infinite-horizon theory [Yu06]. Some introductory references can be found in [Ast65, SS73, TBF05] (the last one contains some simple examples which can help visualization, but the discussion as typos), and the NIPS and AI conferences have regularly papers devoted to POMDPs.

Our goal is to describe the DP algorithm for POMDPs. Essentially, we have already done the work in section 5.1, but it is useful to see the details of the algorithm (5.7),(5.10), in the specialized setting. Suppose then that we have finite number n of possible states (at each stage), and assume without loss of generality that $\mathsf{X}_k = \mathsf{X} = \{1, \dots, n\}$ for all k. The distribution of the state x_k given the observations $\mathcal{I}_k$ is given by a vector of n real numbers $p_k = [p_k^1, \dots, p_k^n]^T$, with p_k belonging to the $n-1$ dimensional simplex

$$p_k \in \Delta_{n-1} = \Big\{ [p^1, \dots, p^n]^T \text{ s.t. } p^i \geq 0 \text{ for all } i \text{ and } \sum_{i=1}^{n} p^i = 1 \Big\}.$$

[2]This version: September 29 2009

[3]In practice, it seems that the performance of the available algorithms has improved significantly in the last decade, so please refer to the literature to know the typical size of the problems that can currently be solved.

Here p_k^i represents the probability that x_k is state i given the available information vector $\mathcal{I}_k$. The vector p_k evolves according to the recursion (5.2):

$$p_{k+1} = \Phi_k(p_k, u_k, y_{k+1}), \tag{5.22}$$

and we mentioned before that it is often called the "belief state". We assume p_0 to be given. Equation (5.7) can now be written

$$\bar{J}_N^*(p_N) = p_N^T c_N \tag{5.23}$$

where c_N is the vector $c_N = [c_N^1, \ldots, c_N^n]^T$ of costs for each possible state at stage N. Similarly, define for stage k for each control u_k the n-dimensional vector $c_k(u_k) = [c_k^1(u_k), \ldots, c_k^N(u_k)]^T$ by

$$c_k^i(u_k) = E_{w_k}[c_k(x_k, u_k, w_k)|x_k = i],$$

as in the definition of $\hat{c}_k(x_k, u_k)$ in (5.9). Then the expected immediate cost at time k in state p_k when choosing control u_k is $p_k^T c_k(u_k)$. The DP recursion is

$$\bar{J}_k^*(p_k) = \min_{u_k \in \mathsf{U}_k} \left\{ p_k^T c_k(u_k) + E_{y_{k+1}} \left[\bar{J}_{k+1}^*(\Phi_k(p_k, u_k, y_{k+1})) \,\middle|\, p_k \right] \right\}.$$

Since we assume the state, control and observation spaces to be finite, all the necessary operations to compute the DP recursion can be performed using matrix calculus. Let us first define some notation. For a given control u at stage k, the state transition matrix P_k^u has elements defined by

$$P_k^u(i,j) = P(x_{k+1} = j | x_k = i, u_k = u).$$

The elements $P_k^u(i,j)$ can be obtained from the knowledge of the distribution of w_k given x_k, u_k (see e.g. (5.5)), or specified directly in a controlled Markov chain formulation. In the latter case we could then have started with a model specifying stage costs of the form $c_k(x_k, u_k, x_{k+1})$, in which case the vector $c_k(u_k)$ above would have been

$$c_k^i(u_k) = \sum_{j=1}^n P_k^{u_k}(i,j) c_k(i, u_k, j).$$

Next define the matrix Q_k^u relating observations to states by

$$\begin{aligned} Q_0(i,o) &= P(y_0 = o | x_0 = i), \\ Q_k^u(i,o) &= P(y_k = o | x_k = i, u_{k-1} = u). \end{aligned}$$

The entries of these matrices can be obtained from the knowledge of the distribution of v_k given x_k, u_{k-1}, or specified directly in the model. First we rewrite explicitly the Bayes filter of section 5.1, i.e., the dynamics (5.22), in terms of these quantities. Recall that starting with the conditional distribution p_k, we

first incorporate the effect of the control u_k in the propagation or time-update step (5.3):

$$\begin{aligned}\overline{p}_{k+1}^i &:= P(x_{k+1}=i|p_k,u_k) = \sum_{j=1}^n P(x_{k+1}=i|u_k,x_k=j)p_k^j \\ &= \sum_{j=1}^n P_k^{u_k}(j,i)p_k^j\end{aligned}$$

and so

$$\overline{p}_{k+1}^T = p_k^T P_k^{u_k}.$$

Then, in the measurement update step (5.4), we take into account the effect of the new measurement y_{k+1}, to get

$$p_{k+1}^i = [\Phi_k(p_k u_k, y_{k+1})](i) = \frac{P(y_{k+1}|x_{k+1}=i,u_k)\overline{p}_{k+1}^i}{\sum_j P(y_{k+1}|x_{k+1}=j,u_k)\overline{p}_{k+1}^j} = \frac{Q_{k+1}^{u_k}(i,y_{k+1})\overline{p}_{k+1}^i}{[\overline{p}_{k+1}^T Q_{k+1}^{u_k}](y_{k+1})}.$$

Here $[\overline{p}_{k+1}^T Q_{k+1}^{u_k}](y_{k+1})$ is the index y_{k+1} of the row vector $\overline{p}_{k+1}^T Q_{k+1}^{u_k}$. Finally, to compute the expectation in the DP recursion, we need to record the distribution $P(y_{k+1}|p_k,u_k)$ that we just used in Bayes' rule. We have

$$\begin{aligned}P(y_{k+1}|p_k,u_k) &= \sum_{i=1}^n P(y_{k+1}|x_{k+1}=i,u_k)\overline{p}_k^i \\ &= [p_k^T P_k^{u_k} Q_{k+1}^{u_k}](y_{k+1}).\end{aligned}$$

So finally, the DP recursion can be written:

$$\begin{aligned}\bar{J}_k^*(p_k) &= \min_{u_k\in\mathsf{U}_k}\left\{p_k^T c_k(u_k) + \sum_o P(y_{k+1}=o|p_k,u_k)\,\bar{J}_{k+1}^*(\Phi_k(p_k,u_k,o))\right\} \\ &= \min_{u_k\in\mathsf{U}_k}\left\{p_k^T c_k(u_k) + \sum_o [p_k^T P_k^{u_k} Q_{k+1}^{u_k}](o)\,\bar{J}_{k+1}^*(\Phi_k(p_k,u_k,o))\right\}.\end{aligned} \tag{5.24}$$

We shall now show a basic property of the value function for POMDPs, namely, it is piecewise linear and concave, i.e., for all $k=0,\ldots,N$, there exist a positive integer m_k and n-dimensional vectors $\alpha_k(1),\ldots,\alpha_k(m_k)$ such that

$$\bar{J}_k^*(p_k) = \min_{j\in\{1,\ldots,m_k\}}\{\alpha_k(j)^T p_k\}.$$

As you might have guessed, we prove the property by backward induction. Now at the last stage $\bar{J}_N^*$ is given by (5.23), so it is linear in the state p_N, hence the property holds trivially with $m_N=1, \alpha_N(1)=c_N$. For the induction step, we

substitute the induction hypothesis for $\bar{J}^*_{k+1}$

$$\begin{aligned}\bar{J}^*_{k+1}(\Phi_k(p_k,u_k,o)) &= \min_j\{\alpha_{k+1}(j)^T\Phi_k(p_k,u_k,o)\}\\ &= \frac{1}{[p_k^T P_k^{u_k} Q_{k+1}^{u_k}](o)}\min_j\left\{\sum_{i=1}^n \alpha^i_{k+1}(j)Q^{u_k}_{k+1}(i,o)\overline{p}^i_{k+1}\right\}.\end{aligned}$$

Now note that the factor $[p_k^T P_k^{u_k} Q_k^{u_k}](o)$ cancels in (5.24) to give

$$\bar{J}^*_k(p_k) = \min_{u_k\in\mathsf{U}_k}\left\{p_k^T c_k(u_k) + \sum_o \min_j\left\{\sum_{i=1}^n \alpha^i_{k+1}(j)Q^{u_k}_{k+1}(i,o)[p_k^T P_k^{u_k}](i)\right\}\right\}. \tag{5.25}$$

The expression (5.25) defines a concave piecewise linear function. To see this, the only potential difficulty involves noting that the sum of two concave piecewise linear functions is again of the same form since

$$\min_i\{a_i^T x + b_i\} + \min_i\{c_i^T x + d_i\} = \min_{i,j}\{a_i^T x + b_i + c_j^T x + d_j\}$$

(i.e., + is distributive with respect to min).

Even though we have a finite dimensional representation of the value function, the number of vectors $\alpha_k(j)$ can increase very quickly with the length of the horizon. Indeed, note that if we have m_{k+1} vectors $\alpha_{k+1}(j)$ to define $\bar{J}^*_{k+1}$ distributing the $\min_j$ inside the $\sum_o$ in (5.25) produces $(m_{k+1})^{|\mathsf{Y}_k|}$ terms and the additional $\min_{u_k}$ multiplies the number of terms in the final minimization by $|\mathsf{U}_k|$, so that we can get up to $m_k = |\mathsf{U}_k|(m_{k+1})^{|\mathsf{Y}_k|}$ vectors at the next step of the DP algorithm. The intractability of the procedure comes from this fact. In practice however, a lot of these vectors $\alpha_k(j)$ are obsolete because they define linear functions that are uniformly bounded by other ones over the simplex. So it is important to prune these vectors at each step in order to solve any problem of reasonable size. If you have to solve a POMDP, the DP algorithm above will not be practical. You should look for a software package implementing a more efficient exact method or an approximate method, or better you can invent a new improved algorithm.