LEQG control formulated as recursive estimation problem

Ramine Nikoukhah* Bernard C. Levy†

Abstract

In this paper, we investigate the possibility of formulating ouput feedback, LEQG control problems as pure ML (maximum likelihood) estimation problems. This formulation allows us to apply powerful recursive estimation algorithms for constructing optimal controllers. This formulation can be used to study a number of related problems such as decentralized control (not considered here) from a new perspective.

1 Introduction

Consider the discrete linear stochastic system

$$x(k+1) = A_k x(k) + B_k u(k) + \nu(k) + M_k v(k), \quad k = 0, 1, ..., N-1, \tag{1.1}$$

$$y(k) = C_k x(k) + N_k v(k), \quad k = 0, 1, ..., N-2, \tag{1.2}$$

where $x(0)$ is a Gaussian random vector with mean x_0 and variance P_0, $\nu(k)$ is a known input sequence and $v(k)$ is a white unit variance zero-mean Gaussian sequence.

We are looking for the controller obtained from

$$\min_{u(k)\in\mathcal{Y}_{k-1}} -(2/\theta)\mathrm{E}\{\exp(-\theta/2 \sum_{k=0}^{N} w(k)^T w(k))\} \tag{1.3}$$

where

$$w(k) = F_k x(k) + G_k u(k) - z(k), \quad k = 0, ..., N-1 \tag{1.4}$$

$$w(N) = Hx(N) - z(N), \tag{1.5}$$

where $z(k)$ is a known sequence (e.g. a trajectory to be followed).

When $\theta > 0$, we have a risk seeking problem [1], $\theta < 0$ corresponds to the risk averse situation and thus the H_∞ problem. In this paper, we consider only the risk seeking case.

2 Formulation as an estimation problem

Let us start with the following result which is the basis of what follows.

Lemma 2.1 *Let $\theta > 0$ and consider the problem*

$$\min_u -(2/\theta)E\{\exp(-\theta/2(w(k)^T w(k)))|y\}$$

where

$$w = Fx + Gu - z \tag{2.1}$$

$$y = Cx + Du + Nv, \tag{2.2}$$

where v is a unit-variance zero-mean Gaussian vector. The optimizing u equals the ML estimate of u based on the following observations:

$$z = Fx + Gu + w\theta^{-1/2} \tag{2.3}$$

$$y = Cx + Du + Nv, \tag{2.4}$$

where w is now to be interpreted as a unit-variance zero-mean Gaussian vector.

Let us now go back to the control problem by considering that we are at the stage $N-1$, i.e., we have already applied $u(0)$ through $u(N-2)$, and we have just observed $y(N-2)$. Then, thanks to Lemma 2.1, to compute $u(N-1)$ we need to solve the xo-graph estimation problem (see [2] for details on xo-graph estimation problem) depicted in Figure 1:

*Inria, BP 105, Le chesnay Cedex, France

†Dept. of Electr. Eng. and Comp. Sc., University of California, Davis, CA 95616, USA

$$
\begin{array}{ccccccccccccccccc}
 & & O_{o,0} & & & & O_{o,1} & & & & & & & & & & \\
 & & | & & & & | & & & & & & & & & & \\
O_i & — & \bar{X}_0 & — & O_{d,0} & — & \bar{X}_1 & — & \cdots & — & O_{d,N-2} & — & X_{N-1} & — & \bar{O}_{d,N-1} & — & \bar{X}_N \\
 & & | & & & & | & & & & & & | & & & & | \\
 & & O_{c,0} & & & & O_{c,1} & & & & & & O_{c,N-1} & & & & O_{c,N}
\end{array}
$$

Figure 1: xo-graph for computing $u(N-1)$

where

$$X_k = \begin{pmatrix} x(k) \\ u(k) \end{pmatrix},$$

$$\bar{X}_k = x(k),$$

$$\bar{O}_{d,k}: \ \nu(k) = -x(k+1) + A_k x(k) + B_k u(k) + M_k v(k),$$

$$O_{d,k}: \ \nu(k) - B_k u(k) = -x(k+1) + A_k x(k) + M_k v(k),$$

$$O_i: \ x_0 = x(0) + D_0 \eta(0),$$

where $\eta(0)$ is a a unit variance zero-mean Gaussian vector and $D_0 D_0^T = P_0$.

$$O_{o,k}: \ y(k) = C_k x(k) + N_k v(k),$$

$$\bar{O}_{o,k}: \ 0 = -y(k) + C_k x(k) + N_k v(k),$$

$$O_{c,k}: \ z(k) = F_k x(k) + G_k u(k) + \theta^{1/2} w(k)$$

$$O_{c,N}: \ z(N) = H x(N) + \theta^{1/2} w(N),$$

where $w(k)$ is to be interpreted as a unit variance zero-mean Gaussian vector sequence independent of $v(j)$.

The xo-graph representation of ML estimation problems is a convenient way of representing the observation equations (the O's) and the unknowns (the X's). Everything on the left side of the O's are supposed known, on the right we have the unknowns.

Operations that we use on observations are $\wedge$ which is the combination (addition) and Ξ the extraction. Combination means simply putting together two observation, and extraction of an unknown from an observation correspond to eliminating all the other unknowns and constructing an observation of smallest size that include only the unknow of interest. If the unknown in question is estimable, extraction corresponds (or is equivalent) to constructing the estimate and the estimation error. For details see [2].

It is easy to see that $\bar{O}_{o,k}$ can be dropped completely as far as estimating x and u is concerned because $y(k)$ which is considered to be unknown appears only there.

The control $u(N-1)$ is thus nothing but the smoothed estimate of $u(N-1)$ which can be obtained as the combination of a forward "Kalman filter" in which in addition to the usual observations we have the cost function interpreted as observations and a backward filter based solely on "cost observations". This amounts to reducing the xo-graph into that of Figure 2 where $O_{f,N-1}$ represents the contribution of all the past observations on X_{N-1} and $O_{b,N-1}$ those of the future.

$$O_{f,N-1} - X_{N-1} - O_{b,N-1}$$

Figure 2: Reduced graph

2.1 Forward filter

The forward filter is actually just a standard Kalman filter in which z's are considered as measurements in addition to y's.

In the xo-language,

$$O_{f,k+1} = \Xi_{x(k+1)}(O_{f,k} \wedge O_{o,k} \wedge O_{c,k} \wedge O_{d,k})$$

where

$$O_{f,0} = O_i.$$

This yields the following filter (we have dropped the subscripts k for the sake of simplifying the expression only):

$$\hat{x}(k+1) = \begin{pmatrix} 0 & 0 & 0 & I \end{pmatrix} \begin{pmatrix} AP_kA^T + MM' & AP_kC^T + MN^T & AP_kF^T & -I \\ M^TN + CP_kA^T & CP_kC^T + NN^T & CP_kF^T & 0 \\ FP_kA^T & FP_kC^T FP_kF^T + \theta I & 0 & \\ -I & 0 & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} -Bu(k) + \nu(k) - A\hat{x}(k) \\ y(k) - C\hat{x}(k) \\ -Gu(k) + z(k) - F\hat{x}(k) \\ 0 \end{pmatrix}$$

with

$$\hat{x}(0) = x_0$$

and P_k, which in some sense can be interpreted as error covariance associated with the filter obtained by the following recursion:

$$P_{k+1} = -\begin{pmatrix} 0 & 0 & 0 & I \end{pmatrix} \begin{pmatrix} AP_kA^T + MM' & AP_kC^T + MN^T & AP_kF^T & -I \\ M^TN + CP_kA^T & CP_kC^T + NN^T & CP_kF^T & 0 \\ FP_kA^T & FP_kC^T FP_kF^T + \theta I & 0 & 0 \\ -I & 0 & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \\ 0 \\ I \end{pmatrix}$$

At $N-1$, the contribution of the past can be summerized as follows:

$$O_{f,N-1} : \hat{x}(N-1) = x(N-1) + D_{N-1}\eta(N-1)$$

where $\eta(N-1)$ is unit covariance and $D_{N-1}D_{N-1}^T = P_{N-1}$.

2.2 Computation of $u(N-1)$

The control $u(N-1)$ can now be computed from

$$O_{b,N-1} = \Xi_{x(N-1),u(N-1)}(O_{c,N-1}, O_{d,N-1} \wedge O_{c,N}),$$

$O_{f,N-1}$ and $O_{c,N-1}$ as follows. Let

$$\Xi_{u(N-1)}(O_{b,N-1} \wedge O_{f,N-1} \wedge O_{c,N-1}) : \\ \hat{u}(N-1) = u(N-1) + K_{N-1}\mu(N-1)$$

then $u(N-1) = \hat{u}(N-1)$ is the optimal control.

2.3 How to compute other u's

The key result is the following

Lemma 2.2 *Consider the ML estimation problem*

$$y - Kx_1 = Lx_2 + Mv$$

where v is unit-variance zero-mean Gaussian and y and x_1 are known, and suppose the solution is

$$Tx_2 = Z(y - Kx_1).$$

Now consider the ML estimation problem for x_1 in the following two cases:

- *Case 1*

$$y = Kx_1 + Lx_2 + Mv$$

- *Case 2*

$$y = Kx_1 + Lx_2 + Mv$$

and

$$Zy = ZKx_1 + Tx_2.$$

Then the ML estimates coincide in the two cases.

In principle, to compute $u(N-2)$ we have to take into account the strategy used at $N-1$, however, thanks to Lemma 2.2, we do not have to. In a sense this result says that at any time k, compute all future controls $u(j)$, $j = k, ..., N-1$, (based solely on information up to $k-1$) and use only $u(k)$. In practice of course we do not actually compute all future u's; we can obtain directly $u(k)$ by constructing the proper backward filter for solving the xo-estimation problem illustrated in Figure 3.

To construct the solution, we can use the backward filter

$$O_{b,k} = X_{x(k),u(k)}(O_{b,k+1} \wedge O_{c,k} \wedge \bar{O}_{d,k}).$$

with $O_{b,N-1}$ as previously defined.

This yields the following backward filter:

$$\check{x}(k) = \begin{pmatrix} 0 & 0 & I & 0 \end{pmatrix} \begin{pmatrix} Q_{k+1} + MM^T & 0 & A & B \\ 0 & \theta I & F & G \\ A^T & F^T & 0 & 0 \\ B^T & G^T & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} \check{x}(k+1) + \nu(k) \\ z(k) \\ 0 \\ 0 \end{pmatrix}$$

with Q_K satisfying

$$Q_k = -\begin{pmatrix} 0 & 0 & I & 0 \end{pmatrix} \begin{pmatrix} Q_{k+1}+MM^T & 0 & A & B \\ 0 & \theta I & F & G \\ A^T & F^T & 0 & 0 \\ B^T & G^T & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \\ I \\ 0 \end{pmatrix}$$

Finally, $u(k)$ can be computed as follows. Let $\Xi_{u(k)}(O_{f,k} \wedge O_{c,k} \wedge O_{b,k}) : \hat{u}(k) = u(k) + K_k \mu(k)$ then $u(k) = \hat{u}(k)$ is the optimal control.

So,

$$u(k) = \begin{pmatrix} 0 & 0 & 0 & I & 0 \end{pmatrix} \begin{pmatrix} P_k & 0 & 0 & I & 0 \\ 0 & Q_{k+1}+MM^T & 0 & A & B \\ 0 & 0 & \theta I & F & G \\ I & A^T & F^T & 0 & 0 \\ 0 & B^T & G^T & 0 & 0 \end{pmatrix}^{-1} \begin{pmatrix} \hat{x}(k) \\ \tilde{x}(k+1)+\nu(k) \\ z(k) \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{tikzcd}
 & O_{o,0} \arrow[d, dash] & & & O_{o,k-1} \arrow[d, dash] & & & & & & \\
O_i \arrow[r, dash] & \bar{X}_0 \arrow[r, dash] \arrow[d, dash] & \cdots \arrow[r, dash] & O_{d,k-2} \arrow[r, dash] & \bar{X}_{k-1} \arrow[r, dash] \arrow[d, dash] & O_{d,k-1} \arrow[r, dash] & X_k \arrow[r, dash] \arrow[d, dash] & \cdots \arrow[r, dash] & X_{N-1} \arrow[r, dash] \arrow[d, dash] & \bar{O}_{d,N-1} \arrow[r, dash] & \bar{X} \arrow[d, dash] \\
 & O_{c,0} & & & O_{c,k-1} & & O_{c,k} & & O_{c,N-1} & & O_{c,N}
\end{tikzcd}$$

Figure 3: xo-graph for computing $u(k)$

3 Conclusion

Preliminary results presented in this paper are a first step towards a complete formulation of LEQG problems in terms of estimation problems and the use of powerful recursive estimation algorithms for constructing their solutions.

References

[1] Whittle, P., Risk-sensitive Optimal Control, 1990.

[2] Nikoukhah, R., D. Taylor, B.C. Levy and A.S. Willsky, "Graph Structure and Recursive Estimation of Noisy Linear Relations," to appear in Journal of Mathematics Systems, Estimation and Control.