Parikshit Shah & Mukul Agarwal

(1) Information patterns: how they matter, using examples from Stochastic Control

(0) Information pattern refers to the data available at various agents, which're responsible to take some action (the action might be estimation, control, etc). for example, in an estimation problem with n sensors & n agents, the following two are examples of information patterns (obviously there are many others):

a. The output of each sensor is known to each agent : this corresponds to centralized estimation

b. for each sensor, there is a corresponding agent who sees that observation; no one else: this corresponds to de-centralized estimation

It is evident that many important problems can be thought of as: "What we can do when the information pattern is the following?". In this report, we consider very simple examples from stochastic control & see that in some cases, the problem is easily solvable, whereas in others, it might be very difficult to solve. We will only look at examples where there is 1 observer & 1 controller; even in this case, we will see that there exist information patterns where the problem of finding the optimal control is difficult to solve.

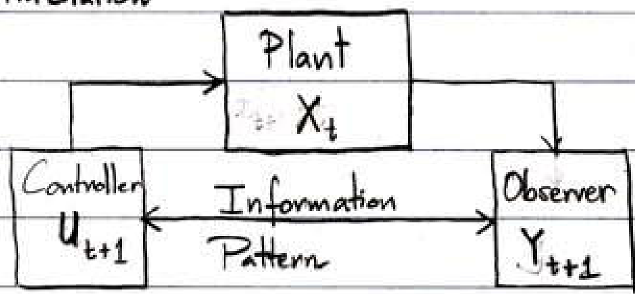
In Section 1, we state the problem formulation of our stochastic control problem. In Section 2, we consider examples of information patterns: classical, strictly classical, non-classical

In Section 3, we make a few definitions: field, field basis, conditioning basis, which will be useful in the later sections. In Section 4, we get into business, proving separation of estimation & control for the classical information pattern, & give an example where separation doesn't hold for the non-classical pattern. In Section 5, we illustrate the importance of the concept of conditioning basis by showing that dynamic programming works because of this (more precise statement in Section 5). Again, for classical information pattern, I_t is a conditioning basis for X_{t-1} (I_t suitably defined; details in Section 5), and this, along with separation shows that dynamic programming works. Again, things break down, in general for non-classical information Patterns. Section 6 Looks at the Witsehausen's counterexample of a 2 step system with non-classical information pattern with squared cost & Gaussian noise & it is proved that there exists a non-affine controller which performs better than the best affine controller (this is important because for the classical information pattern, it can be proved that there exists an optimal affine controller); infact the optimal controller for this problem hasn't been found yet. In Section 7, we conclude.

It might be more useful to look at this list of "which section corresponds to what" after reading Sections 1 & 2, where we make all the important definitions required to understand what is written above.

(1) <u>Problem Formulation (same as Witsenhausen [1] "Separation of Estimation & Control for discrete time systems"</u>, with $M = K_i = 1$.

Consider a system evolving for T time steps. Observations 're made at 1 observation post & control inputs 're applied at each step from 1 control station



The operation of the system can be described chronologically as follows:

Generation of initial random state X_0
Observation of output Y_1
Application of control input U_1
Transition to state X_1
...
Transition to state X_{t-1}
Observation of output Y_t
Application of control input U_t
Transition to state X_t
...
Transition to state X_T

$$\boxed{X_0 \; Y_1 \; U_1 \; X_2 \; Y_2 \; \cdots \; X_{t-1} \; Y_t \; U_t \; X_{t+1} \cdots \; X_T}$$

The transition functions:

The observation y_t from x_t is given by the function $\boxed{Y_t = g_t(X_{t-1}, W_t}$ where W_t is a noise random variable (observation noise at time t)

If a control U_t is applied, the state of the system, $\boxed{X_{t+1} = f_t(X_{t-1}, V_t, U_t)}$, where V_t is a noise random variable (controller noise at time t).

Primitive random variables:

$X_0 \rightarrow$ initial state

$W_t, 1 \leq t \leq T \rightarrow$ Observation noise

$V_t \rightarrow$ Controller noise

We assume that $\boxed{X_0; W_t, 1 \leq t \leq T; V_t, 1 \leq t \leq T}$ are independent random variables with given distributions.

Cost:

$$C = \sum_{i=1}^{T} h_t(x_t, u_t)$$

Aim: To minimize expected cost.

We assume that X_0, W_t, V_t take values in a finite set. We make this assumption, only to circumvent some measure-theoretic language of measurable functions, etc (in the section on Witsenhausen's Counterexample, we lift this assumption, since that example considers Gaussian Noises).

The Control Law:

We need to specify how we determine the control action. We need to specify 3 things:

a. Data available as arguments of the law.

b. Restriction on values taken.

c. Restrictions on functional form of law.

a. Data available as arguments of the law

This is what is the information pattern. At time t, the controller will have access to $I_t \subseteq \{y_1, \dots, y_t, u_1, \dots, u_{t-1}\}$.

The control at time t,

$$\boxed{u_t = \gamma_t(I_t)}$$

There's another possibility: controller also has access to functional forms of past control laws.

$$\boxed{u_t = \gamma_t(I_t, \gamma_1^{t-1})}$$

Note the distinction between:

knowing the past control law (eg. knowing u_1, u_3, u_7)

and

knowing the functional form of past control law (eg. knowing $\gamma_1, \gamma_3, \gamma_7$)

Allowing access to past control law functional forms doesn't lead to any difference. This is because, while calculating optimal control laws, let's assume $U_t = \gamma_t(I_t, \gamma_1^{t-1})$. Let the optimal control laws be $\gamma_1^*, \dots, \gamma_T^*$. Now, when we're running the system, $U_t = \gamma_t^*(I_t, \gamma_1^{*t-1})$, but γ_1^{*t-1} are fixed & hence, u_t is actually a function just of I_t, $U_t = \gamma_t^*(I_t)$! Hence, while running the system, we don't need to remember the functional forms of control laws!

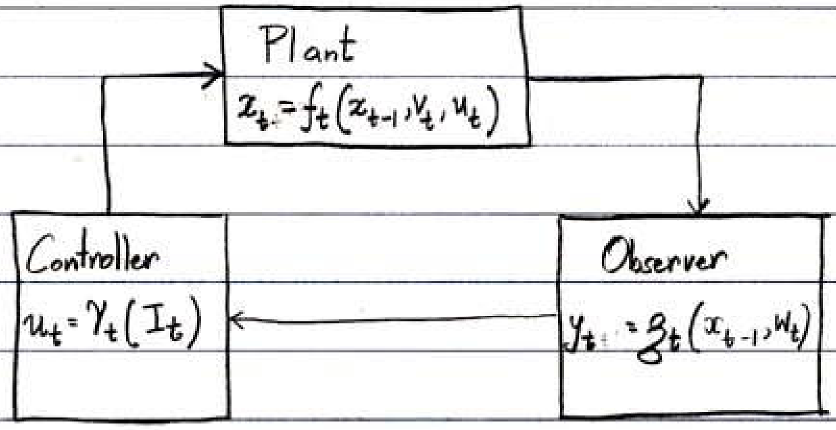
b. **Restrictions on the values taken (the range):**

We assume that the control applied at each time t lies in a finite set. Again, we make this assumption, only to make the presentation of important concepts simpler. This assumption is lifted in the section on Witsenhausen's counter-example.

c. **Restrictions on the functional form of the law**

for example, the control law may need to be affine. We will look at this in the section on Witsenhausen's counter-example.

AIM: To design γ_t, $1 \le t \le T$, so as to minimize the expected cost.



(2) Examples of Information Patterns:

a. Strictly Classical Information Pattern:

All past information (observations & control) available.

$I_t = \{ y_1^t, u_1^{t-1} \}$. As well shall see, this is the easiest case to handle. Infact, even if we had had access to functional forms of past control laws, that is, even if we were allowed $u_t = \gamma_t(I_t, \gamma_1^{t-1})$, it wouldn't have made a difference to the optimal cost.

b. Classical Information Pattern:

We have perfect recall: $I_t \subseteq I_{t+1}$.

Note that if the controller fails to record it's observations, it might still have perfect recall. for example, I_t might just be $I_t = \{y_1, \dots, y_t\}$. Also, we said at the end of the last section that we can always assume access to functional forms of past control laws when calculating the optimal control law. Because of perfect recall, this means that at any time t, we have access to all the past controls (another way of saying this is that because of perfect recall, even if we have forgotten past control laws, we can reconstruct the past control actions from the past observations & past functional forms of control laws).

c. General Information Pattern:

As the name suggests, any non-classical information pattern would be called a general information pattern. As an example, consider a two step problem ($T=2$):

$$T = 2$$
$$I_1 = \{y_1\}$$
$$I_2 = \{y_2\}$$

Thus, in the second step, the observation & control action of first step have been forgotten.

We will be considering the strictly classical, classical & the above example of non-classical information pattern recurrently in this report, & see that the first two information patterns are easy from point of view of understanding properties of the optimal control law & calculating the optimal control law, whereas the last one is difficult from both these points of view. Infact, Witsenhausen's counter-example has the last information pattern above as it's information pattern.

Before we get there, we state a few VERY IMPORTANT definitions, which 'll be relevant in any problem where there are information patterns.

(3) <u>Definitions:</u>

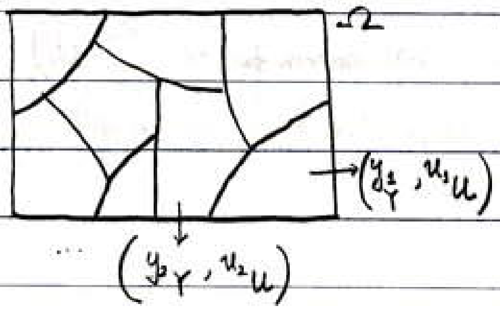
Note that unless the functional forms of control laws $\gamma_1, \dots, \gamma_t$ are specified, we cannot talk about conditional distributions $P_{x_{t-1}|I_t}$. These are not well defined, and depend on the control law functional forms γ_1^t.

<u>Defn (The space of primitive random variables):</u>

$\Omega = (X_0, V_1^T, W_1^T)$

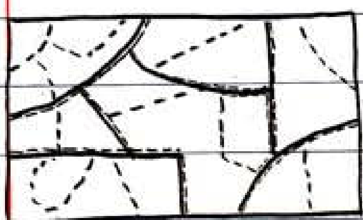
<u>Defn (Field):</u>

Given the functional forms of the control laws γ_1^T. Let $Y \subseteq \{Y_1^T\}$, $U \subseteq \{U_1^T\}$. A vector of observations on set Y will be denoted by y_Y. A vector of controls restricted to U will be denoted by u_U. Given $(y_Y, u_U)\ \gamma_1^T$, we can pin-point the subset of Ω which led to these observations. <u>This subset will depend on γ_1^T</u>. We call this partition of Ω, the field generated by the observations Y, U. We denote it by $\mathfrak{F}(Y, U; \gamma_1^T)$.



Example:

Recall that classical information pattern has perfect recall. In other words, we have more information at time $t' > t$. It follows that $\mathcal{F}(I_t ; \gamma_1^T)$ is a finer partition as compared to $\mathcal{F}(I_{t-1}, \gamma_1^T)$. Another way of saying this is that the fields are nested



The dotted partition $\rightarrow \mathcal{F}(I_t ; \gamma_1^T)$
The bold partition $\rightarrow \mathcal{F}(I_{t-1}, \gamma_1^T)$
Any set in $\mathcal{F}(I_t ; \gamma_1^T)$ is a subset of some set in $\mathcal{F}(I_{t-1} ; \gamma_1^T)$

Defn (Field Basis at time t):

We say that $(Y, U) \subseteq (Y_1^t, U_1^t)$ is a field basis at time t, if, for any two designs γ_1^t and $\hat{\gamma}_1^t$ WHICH ARE COMPATIBLE WITH THE INFORMATION PATTERN, $\mathcal{F}(Y, U; \gamma_1^t) = \mathcal{F}(Y, U; \hat{\gamma}_1^T)$.

Example:

For the perfectly classical information pattern, $(Y, U) = I_t = (Y_1^t, U_1^{t-1})$ is a field basis at time t. We will prove this fact in a later section.

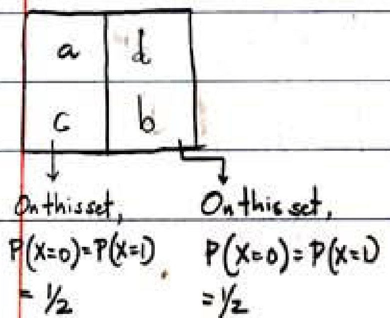
Defn (Conditioning Basis at time t):

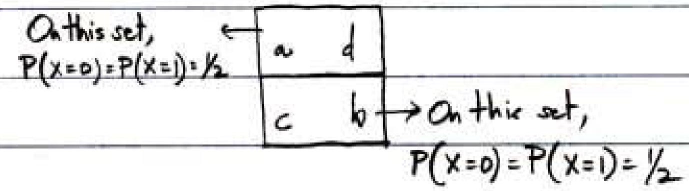
Many times, what matters, is not the fields being the same, but the conditional probabilities being the same.

We say that (Y, U) is a conditioning basis for a random variable X (for example $X = X_t$) if for any two designs γ_1^T & $\hat{\gamma}_1^T$ compatible with the information pattern,

$$P_{X|(Y,U,\gamma_1^T)} = P_{X|(Y,U,\hat{\gamma}_1^T)}$$

Example: If (Y, U) is a field basis at time t, (Y, U) is a conditioning basis for the state X_{t-1}. This is because fields are the same, so are the conditional distributions. But the converse is not true.
It can always happen that Ω gets partitioned in a different way, but conditional distributions are still the same. To illustrate this point, we consider a simple example from probability (not in this setting).
Let $\Omega = \{a, b, c, d\}$, each with probability $1/4$, & X be a random variable, $X(a) = X(b) = 1$; $X(c) = X(d) = 2$. Partition of Ω into sets $\{a, c\}$ & $\{b, d\}$ results in $P_{X|\{a,c\}}(1|\{a,c\}) = P_{X|\{a,c\}}(2|\{a,c\}) = P_{X|\{b,d\}}(1|\{b,d\}) = P_{X|\{b,d\}}(2|\{b,d\}) = 1/2$. Partitioning of Ω into sets $\{a, d\}$, $\{b, c\}$ also results in same probabilities.





Thus, being a field basis is a stronger condition than being a conditioning basis for a random variable.

From the point of view of stochastic control though, it seems that if the information pattern I_t is a conditioning basis for X_{t-1}, it considerably simplifies the calculation of optimal policy. We will come to this point later, in Section 5.

4 Results concerning separation of estimation & control :

Definition (Separated actions): If each γ_t can be written as $\gamma_t = F \circ P_{X_{t-1} | I_t, \gamma_1^{t-1}}$, we say that the control laws're separated.

This means that we can decompose our control laws into two steps:

a. Compute distribution of X_{t-1} given past knowledge (I_t, γ_1^{t-1})

b Apply a control depending on this conditional distribution & not explicitely depending on past knowledge.

This has intuitive appeal because we have "separated" our control laws into "estimation" & "control".

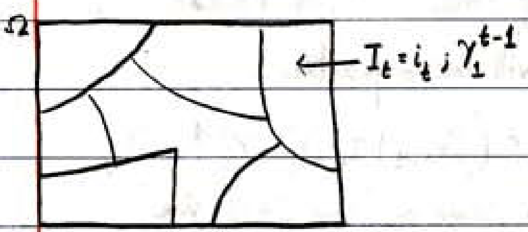
Definition (Separation holds): We say that separation holds if there exist γ_1^t separated such that the expected cost is minimized

Another way of saying this is that $P_{X_{t-1} | I_t, \gamma_1^{t-1}}$ is a sufficient statistic from the point of view of minimizing the expected cost.

<u>Theorem:</u> Separation holds for the classical information pattern.

<u>Proof:</u>

Assume that control laws γ_1^{t-1} have been used from time 1 to $(t-1)$. We prove that γ_t^T are separated. To this end, assume that the system has evolved upto time $(t-1)$ & at time t, the information pattern I_t has taken the value i_t. Now, based on the knowledge of i_t and γ_1^{t-1}, we can pin-point, which subset of Ω has occured. Call this subset A_{i_t} (we supress γ_1^{t-1} since we fix the past actions).



An important observation is that since the information pattern is classical & our knowledge of Ω only increases with time, at times $t' > t$, we always know that A_{i_t} occured (infact, we will, with increasing time be able to pin-point a subset of A_{i_t} as having occured).

As a result, we can consider a new problem on each A_{i_t} for the remaining time $(T-t)$. Note that here we crucially use the fact that information patterns are nested; if they weren't, we wouldn't be able to consider the problem on each A_{i_t} as an independent problem, independent of another observation i'_t at time t. This is because at a large time s, suppose i_t is compatible with i_s & i'_t is compatible with i'_s. If fields're not nested, $A_{i_t} \cap A_{i'_s}$ need not be the empty set. In other words, action taken for the observation i_t at time t can affect the performance on the set $A_{i'_t}$ although the effects will be visible only at a later time, for example s. This does not happen though, for the case of nested information patterns, our knowledge only increases & we can consider the problem on each A_{i_t} as an independent problem for the remaining time $T-t$.

Consider two observation i_t & i'_t at time t & the corresponding sets A_{i_t} & $A_{i'_t}$. for the remaining time $T-t$, it is clear that the expected cost on these sets will be a function $c(p_{x_{t-1}|I_t=i_t,\gamma_1^{t-1}}, \gamma_t^T)$ and $c(p_{x_{t-1}|I_t=i'_t,\gamma_1^{t-1}}, \gamma_t^T)$, and hence, by choosing γ_t^T to be the same on these sets, we can make the remaining costs on these sets equal. Also, if γ_t^T gets chosen in such a way that remaining expected cost on one of these sets is larger than the other, we can change the control law on the set with higher expected remaining cost to the set with lower remaining expected cost, & in this process, lower the remaining expected cost (note that all the time, we're using the assumption that we can consider problems on A_{i_t} & $A_{i'_t}$ for remaining time as independent problems.)

It follows that if i_t & i'_t are such that $p_{x_{t-1}|I_t=i_t,\gamma_1^{t-1}} = p_{x_{t-1}|I_t=i'_t,\gamma_1^{t-1}}$, we can choose the control laws for remaining time to be the same for both. In other words, control law at time t only depends on $p_{x_{t-1}|I_t=i_t,\gamma_1^{t-1}}$

This holds for all time t. It follows that separation holds.

Comments:

a. We crucially used the fact that information patterns're nested; we used this to assert that we can consider problems on A_{i_t} & $A_{i'_t}$ as independent of each other.

b. We never used any existence of conditioning basis. As we'll see in next section, by choosing I_t appropriately, we can make I_t into a conditioning basis. But the important point here is that this fact is not relevant to the proof of separation for the classical information pattern.

Let us now consider an example with general information pattern; the same example which we gave in Section 2,

$$I_1 = \{y_1\}$$
$$I_2 = \{y_2\}$$
$$T = 2$$

That is, a 2 step problem where the observation & control in first step get forgotten at the second step.

Let $I_1 = i_1 = y_1$ & $I_1 = i_1' = y_1'$ be two possible observations at time 1 such that $P_{X_0|I_1=i_1} = P_{X_0|X_1=i_1'}$. Then, if separation holds, laws corresponding to observations i_1 and i_1' in the first step would be the same. But it is not clear that separation holds at first step. In words of John Tsitsiklis, since y_1 gets forgotten at step 2, the

<u>controller might somehow want to convey the knowledge of y_1 to the second step</u>, which means that controls corresponding to y_1 & y_1' won't be the same. Another way of saying this is that the controller would want to <u>code for y_1</u>.

Thus, we don't expect separation to hold in general.

But why is separation an important criterion in the first place. In the next section, we look at another criterion: computing the optimal control law, & see the relevance of the concept of conditioning basis.

6. Witsenhausen's Counterexample In this section we introduce the well known Witsenhausen counterexample [Witsenhausen2]. The counterexample is a simple discrete time stochastic control problem over two stages. The dynamics of the system are linear, the noise is Gaussian, the cost is a quadratic function of the state and inputs, yet the problem is surprisingly hard to solve. It is sometimes erroneously believed that under this setting, the problem always has optimal control laws with affine structure. We will prove that this is in fact not the case using the approach in [MitterSahai]. It is appropriate to mention at this point that the emphasis in this section is *not* separability of estimation and control (the counterexample does satisfy separability). The counterexample is meant to illustrate that nonclassical information patterns lead to difficult optimal control problems where fundamental techniques like dynamic programming fail. (As an interesting aside, the optimal controller for Witsenhausen's counterexample is not yet known, and is an open problem. In fact, in the original paper [Witsenhausen2], considerable effort is spent in proving just the existence of optimal control laws.)

6.1 Formulation Let x_0 be a randomly generated initial state with Gaussian distribution with zero mean and variance σ^2. Let v, the noise variable, be a random variable independent of x_0 with Gaussian distribution with zero mean and unit variance. The system evolves according to the following dynamics:

$$\begin{aligned} y_1 &= x_0 \\ x_1 &= x_0 + u_1 \\ \\ y_2 &= x_1 + v \\ x_2 &= x_1 + u_2 \end{aligned} \tag{1}$$

The control laws are constrained to be of the form:

$$\begin{aligned} u_1 &= \gamma_1(y_1) \\ u_2 &= \gamma_2(y_2) \end{aligned} \tag{2}$$

The cost is given by:

$$J = \mathbf{E}\left[k^2 u_1^2 + x_2^2\right]. \tag{3}$$

Let us examine the formulation of the problem more closely. There is a single input station, and a single output post. As mentioned earlier, the dynamics are linear, the noise is Gaussian, and the cost is quadratic. The first stage output y_1 is a perfect observation of the initial state x_0. However, the problem is not in the standard LQG setup because the input at the second stage does not have direct access to the state, rather it only has access to a noisy state measurement (partial information). The second stage cost could be driven to zero if x_2 could be driven to zero. This would require u_2 to form a perfect estimate of the state x_1. This is of course not possible, since u_2 can only depend on y_2, a noisy version of x_1. Hence the role of u_2 is estimation. The control designer, aware of this difficulty, would try to encode the state x_1 efficiently using the input u_1. Hence the first stage can be thought of as an encoder, with associated coding cost, and the second stage can be thought of as an estimator with the associated cost being the mean square estimation error.

In terms of the language introduced earlier, what is the information pattern corresponding to this problem?

$$Y_1 = \{(1,1)\}$$
$$U_1 = \emptyset$$

$$Y_2 = \{(2,1)\}$$
$$U_2 = \emptyset.$$

The important observation to make here is that the information pattern fails to satisfy perfect recall. Hence the information pattern is *nonclassical*. If the information pattern were classical (i.e. if input y_1 were remembered by the system at the second stage, and $u_2 = \gamma(y_1, y_2)$), then the solution to the problem would be trivial. Indeed, one could set $u_1 = 0$, and $u_2 = -y_1 = -x_1$ giving a policy with $J = 0$.

Nonclassical information patterns can affect analysis and design of systems in two *completely different* adverse ways: (1) Separation may fail, and (2) Dynamic programming may fail. It is interesting to note that separation of estimation and control holds for this example. In the first stage, perfect knowledge of x_0 is available, since $y_1 = x_0$ almost surely. Since the control law is constrained to be of the form $u_1 = \gamma_1(y_1) = \gamma_1(x_0)$, separation holds for the first stage. As commented earlier, the optimal action in the second state is estimating the state x_1 given the input y_2. Hence separation holds for the second stage also. Hence, it is sometimes possible to have separation for nonclassical systems. Finally, we would like to point out that the optimal solution for this problem cannot be determined via dynamic programming.

6.2 Best Affine Controller In this section we compute the best affine controller, which is of the form:

$$u_1 = ay_1 + c_1$$
$$u_2 = by_2 + c_2.$$

It is not hard to see that since the primitive random variables are zero mean, $c_1 = c_2 = 0$. Due to the dynamics of the system, $x_1 = (1+a)x_0$, and thus x_1 is a Gaussian random variable with zero mean and variance $(1+a)^2\sigma^2$. Also, the first stage cost is $\mathbf{E}[a^2x_0^2] = k^2a^2\sigma^2$.

The second stage cost is the mean square error $\mathbf{E}[(x_1 - (-u_2))^2]$ between the state x_1 and its estimate formed by the input u_2. The optimal choice for u_2 is MMSE estimate:

$$u_2(y_2) = -\mathbf{E}\left[x_1|y_2\right] = -\frac{(1+a)^2\sigma^2}{1+(1+a)^2\sigma^2}y_2.$$

The corresponding mean square error is:

$$\mathbf{E}[x_2^2] = \frac{(1+a)^2\sigma^2}{1+(1+a)^2\sigma^2}.$$

The total cost for a fixed a is:

$$J(a) = k^2a^2\sigma^2 + \frac{(1+a)^2\sigma^2}{1+(1+a)^2\sigma^2}. \tag{4}$$

To find the optimal design, one should optimize over a to minimize (4) by setting $\frac{dJ(a)}{da} = 0$.

6.3 Quantizing Controller In this section we show a way to construct nonlinear controllers through quantization as described in [MitterSahai]. The problem described in (1) is parameterized by k and σ^2. We show that in a certain regime of these parameters, the nonlinear controllers perform arbitrarily better than the best affine controllers. Consider a controller of the form:

$$\begin{aligned}\gamma_1(y_1) &= -y_1 + B\lfloor \tfrac{y_1}{B} + \tfrac{1}{2} \rfloor \\ \gamma_2(y_2) &= B\lfloor \tfrac{y_1}{B} + \tfrac{1}{2} \rfloor.\end{aligned}$$

The first stage looks at y_1 and chooses the input such that the state x_1 is quantized into a bin of size B (the input is simply the quantization error). The decoding rule is simply to look at the bin that $y_2 = x_1 + v$ falls in (thus, if the noise were small, it would always choose the correct bin). As the bin size increases, the probability of decoding error decreases exponentially according to the tail of the distribution for v (which is a standard normal distribution).

Consider the series of parametric problems given by:

$$\begin{aligned}{}_n &= \tfrac{1}{n^2} \\ \sigma_n &= n^2 \\ B_n &= n.\end{aligned}$$

For each problem in this sequence, the first stage input is bounded by the maximum quantization error $B_n/2$, and hence the cost is bounded by $\frac{k_n^2 B_n^2}{4} = \frac{1}{4n^2}$. Hence as n tends to infinity, the first stage cost approaches zero. For the second stage, notices that the bin size increases as n, while the variance of noise v is fixed at 1. The estimation error is zero unless the noise has magnitude larger than $\frac{B}{2} = \frac{n}{2}$. Since v is Gaussian, this tail event happens with a probability approaching zero. Thus, in the limit of large n,

$$\lim_{n\to\infty} J_n = 0.$$

For this same sequence, it is easy to see that the cost for the optimal affine controller approaches 1, and hence the quantizing controller performs arbitrarily better.

References:

[Witsenhausen1] Separation of Estimation & Control for Discrete time systems by H. S. Witsenhausen, IEEE Tran Info. Th., Nov 1971, Vol 59, Issue 11

[Witsenhausen2] A Counterexample in Stochastic Optimum Control, Siam J. Control Vol 8, No 1, 131-147, 1968

[Witsenhausen3] On Policy Independence of Conditional Expectation by H.S. Witsenhausen, Information & Control, Vol 28.

[MitterSahai] Information & Control: Witsenhausen Revisited by S. Mitter & A. Sahai, 1997