Variable Time Delay Estimation in Colored and Correlated Noise Environment

NADER M. NAMAZI, Senior Member, IEEE
The Catholic University of America
BHARAT BISWAL
Michigan Technological University

A new iterative algorithm for variable time delay estimation (TDE) is presented. The algorithm is based on a system model where two received signals of the form

\[ r_1(i) = s(i) + n_1(i) \]
\[ r_2(i) = s(i - D(i)) + n_2(i) \]

are used to estimate the delay. In the model \( i \) is the discrete-time index, \( s(i) \) is the source signal, \( D(i) \) is the (integer) delay waveform, and \( n_1(i) \) and \( n_2(i) \) are colored and correlated noises.

Based on the minimum mean-squared error criterion and a linearized observation model, we devise an iterative delay estimator. The estimator involves a weighting function that is designated to facilitate the convergence to delay estimate.

Simulations are run to test the performance of the algorithm. Results of the simulations are presented.

\[ D(i) \text{ and } D \]
\[ \hat{D}(i) \text{ and } \hat{D} \]
\[ s(i) \text{ and } \tilde{s} \]
\[ \hat{s}(i) \text{ and } \hat{\tilde{s}} \]
\[ \nabla \]
\[ G \]
\[ r(i) \text{ and } r_2 \text{, } \ell = 1, 2 \]
\[ r_{2D} \text{ and } r_{2D} \]
\[ \hat{r}(i) \text{ and } \hat{\hat{r}} \]
\[ n_1(i) \text{ and } n_2 \text{, } \ell = 1, 2 \]
\[ n_{2D}(i) \text{ and } n_{2D} \]
\[ \hat{n}(i) \text{ and } \hat{\hat{n}} \]
\[ K_{D}(i, j) \text{ and } K_D \]
\[ K_s(i, j) \text{ and } K_s \]
\[ \hat{R}_D \]
\[ \hat{R}_f \]
\[ \hat{R}_n \]
\[ R_{n1} \text{ and } R_{n2} \]
\[ R_{12} \]
\[ C_D \text{ and } C_D \]
\[ C_s \text{ and } C_s \]
\[ m, M \]
\[ m', M' \]
\[ i, I \]

II. INTRODUCTION

Variable time delay estimation (TDE) is a major problem of concern in radar and sonar systems. In these systems, the location of a moving target can be identified from the measure of the variable time delay signal. In this paper, it is assumed that a
strategy in \( m \) direction; that is, for a given \( m \), the estimator manipulates \( \hat{D}^{(m+1)}(i) \) to obtain \( \hat{D}^{(m+1)}(i) \) for all \( i \). The correction term includes the difference signal \( r_1(j) - r_2(j + \hat{D}^{(m)}(j)) \) which contains the delay estimation error; this useful error term tends to drive the algorithm into lock.

There is an important virtue associated with the algorithm (3), namely, the incorporation of the weighting function \( K_D^{(m+1)}(i, j) \). As presented in Subsection A, this function is designated based on the first-order approximation of the algorithm (3). The role of the function is to facilitate the convergence of the algorithm to the desired results in colored and correlated noise environment.

In the subsequent sections, we utilize the following assumptions.

1. \( \hat{D}^{(0)}(i) = \mathcal{E}\{D(i)\}, \forall i \).
2. The noises \( n_1(i) \) and \( n_2(i + \hat{D}^{(m)}(i)) \) are zero mean.
3. 

\[
\mathcal{E}\{D(i)n_1(j)\} = \mathcal{E}\{D(i)\}\mathcal{E}\{n_1(j)\}
\]

and

\[
\mathcal{E}\{D(i)n_2(j + \hat{D}^{(m)}(j))\} = \mathcal{E}\{D(i)\}\mathcal{E}\{n_2(j + \hat{D}^{(m)}(j))\}
\]

Note that this assumption in conjunction with the assumption 2 implies that \( \mathcal{E}\{D(i)n_1(j)\} = 0 \) and \( \mathcal{E}\{D(i)n_2(j + \hat{D}^{(m)}(j))\} = 0, \forall i, j, m \).

In the following subsection, we establish an iterative method by which the function \( K_D^{(m+1)}(i, j) \) is derived. The procedure is based on the minimum mean square error (MMSE) criterion.

A. Derivation of \( K_D^{(m+1)}(i, j) \)

We now develop a method to obtain the function \( K_D^{(m+1)}(i, j) \). The procedure exploits the first-order approximation of (3) and is parallel to the calculation of the Kalman gain [10].

To facilitate the developments, we write (3) in vector-matrix form, that is,

\[
\hat{D}^{(m+1)} = \hat{D}^{(m)} + K_D^{(m+1)} \hat{r}^{(m)}
\]

where \( \hat{D} = [\hat{D}(1) \hat{D}(2) ... \hat{D}(I)]^T \), in which \( T \) denotes the transpose of the vector, \( K_D^{(m+1)} = [K_D^{(m)}(i, j)]; i, j = 1, 2, ..., I \),

\[
\hat{r}^{(m)} = r_1 - r_2^{(m)}
\]

in which \( r_1 = [r_1(1) r_1(2) ... r_1(I)]^T \), and \( r_2^{(m)} = [r_2(1 + \hat{D}^{(m)}(1)) r_2(2 + \hat{D}^{(m)}(2)) ... r_2(I + \hat{D}^{(m)}(I))]^T \).

Expression (5) can be approximated as

\[
\hat{r}_L^{(m)} = \mathbb{G}(D - \hat{D}^{(m)}) + \hat{n}^{(m)}
\]

where \( \hat{r}_L^{(m)} \) is the first-order approximation of \( \hat{r}^{(m)} \), \( \mathbb{G} = \text{diag} \{\nabla s(1) \nabla s(2) ... \nabla s(I)\} \) in which \( \nabla \) signifies
the backward difference operator defined as \( \nabla s(i) = s(i) - s(i-1) \),
\[
\tilde{n}(m) = n_1 - n_{2D}^{(m)}
\]  
where \( n_1 = [n_1(1) \, n_1(2) \ldots n_1(I)]^T \), and \( n_{2D}^{(m)} = [n_{2D}^{(m)}(1) \, n_{2D}^{(m)}(2) \ldots n_{2D}^{(m)}(I)]^T \) in which \( n_{2D}^{(m)}(i) = n_2(i + \hat{D}(m)(i)) \). It follows from (4) and (5) with \( \hat{D}(m) = \hat{D}_L(m) \) that
\[
\hat{D}(m+1) = (I - K^{(m+1)}_D)\hat{D}(m) - K^{(m+1)}_D \tilde{n}(m)
\]  
wherein \( \hat{D}(m) = D - \hat{D}(m) \) and 1 denotes the \( I \times I \) identity matrix. Assumption 2 implies that \( \mathcal{E}\{\tilde{n}(m)\} = 0 \). Also, in view of the assumption 1, it is seen that \( \mathcal{E}\{\tilde{D}(0)\} = 0 \). Henceforth, from (8) it follows that when \( \hat{D}(m) = \hat{D}_0(m) \), the estimates \( \hat{D}(m) \) are unbiased.

To proceed further, consider the covariance matrix of the error vector \( \hat{D}(m+1) \) denoted as \( \tilde{R}_n^{(m+1)} \). This matrix can be obtained by taking the expected value of the multiplication of (8) with its transpose; henceforth,
\[
\tilde{R}_D^{(m+1)} = (I - K^{(m+1)}_D \hat{D}(m)) \tilde{R}_D^{(m+1)} [I - K^{(m+1)}_D \hat{D}(m)]^T + K^{(m+1)}_D \tilde{R}_n^{(m+1)} K^{(m+1)T}_D
\]  
where \( \tilde{R}_n^{(m)} \) is the covariance matrix of \( \tilde{n}(m) \) and the fact that \( \hat{D}(m) \) and \( \tilde{n}(m) \) are orthogonal has been utilized. The latter follows from assumptions 2 and 3.

We determine the matrix \( K^{(m+1)}_D \), \( \forall m \), in such a manner for which the trace of the error covariance matrix \( \tilde{R}_D^{(m+1)} \) is minimum; that is, we seek for the weighting matrix that minimizes the mean-squared error. A necessary condition is obtained if we differentiate the trace of the matrix \( \tilde{R}_D^{(m+1)} \) in (9) with respect to \( K^{(m+1)}_D \) and set the result equal to zero. Therefore,
\[
K^{(m+1)}_D = \tilde{R}_D^{(m)G}[G\tilde{R}_D^{(m)G} + \tilde{R}_n^{(m)}]^{-1}
\]  
where it is assumed that the matrix \( \tilde{R}_n^{(m)} \), \( \forall m \), is positive definite to ensure the existence of the inverse matrix in (10). Combination of (9) and (10) results in a more convenient form for computation of \( \tilde{R}_D^{(m+1)} \) as shown below:
\[
\tilde{R}_D^{(m+1)} = [I - K^{(m+1)}_D \hat{D}(m)] \tilde{R}_D^{(m)}. 
\]  
Note that the assumption 1 implies that \( \tilde{R}_D^{(0)} = C_D \), where \( C_D = [C_D(i,j)] \), \( i, j = 1, 2, \ldots, I \), is the covariance matrix of \( D \).

It should be emphasized that the algorithm is capable of tracking the delay estimate if the delay estimation error is relatively small. Otherwise, the first-order approximation of the difference signal \( \hat{D}(m) \) is no longer valid which, consequently, results in inaccurate calculation of the weighting matrix \( K^{(m+1)}_D \).

Therefore, two modes of action of the algorithm should be characterized, namely, the initialization (acquisition) mode and the tracking mode. It is crucial that in the initialization mode of the algorithm, the delay error, \( \hat{D}(0) \), be within some limit or else the algorithm may not proceed into the tracking mode. This fact is further explored in the simulation experiments.

In the following subsection, we obtain the covariance matrix \( \tilde{R}_n^{(m)} \).

### 8. Derivation of \( \tilde{R}_n^{(m)} \)

Here we evaluate the covariance matrix of the noise \( \tilde{n}(m) \). It follows from (7) that
\[
\tilde{R}_n^{(m)} = R_n + \mathcal{E}\{n_{2D}^{(m)} n_{2D}^{(m)T}\} - \mathcal{E}\{n_1 n_{2D}^{(m)T}\} - \mathcal{E}\{n_1 n_{2D}^{(m)T}\}^T
\]  
where \( R_n = [R_n(i,j)] \), \( i, j = 1, 2, \ldots, I \) signifies the covariance matrix of \( n_1 \).

Using the linear interpolation depicted in Fig. 1, we obtain
\[
n_{2D}^{(m)}(i) = (1 - \delta^{(m)}(i))n_2(\ell^{(m)}(i)) + \delta^{(m)}(i)n_2(\ell^{(m)}(i) + 1)
\]  
where \( \delta^{(m)}(i) = \hat{D}(m)(i) - \ell^{(m)}(i) \). Henceforth, from (13), we obtain
\[
\mathcal{E}\{n_{2D}^{(m)}(i) n_{2D}^{(m)T}(j)\} = (1 - \delta^{(m)}(i))(1 - \delta^{(m)}(j))R_{12}(\ell^{(m)}(i), \ell^{(m)}(j))
\]
\[
+ \delta^{(m)}(i)(1 - \delta^{(m)}(j))R_{12}(\ell^{(m)}(i), \ell^{(m)}(j) + 1) + \delta^{(m)}(i)(1 - \delta^{(m)}(j))R_{12}(\ell^{(m)}(i) + 1, \ell^{(m)}(j))
\]
\[
+ \delta^{(m)}(i)\delta^{(m)}(j)R_{12}(\ell^{(m)}(i) + 1, \ell^{(m)}(j) + 1).
\]  
Correspondingly, from (13) the \( (i,j) \) elements of the matrix \( \mathcal{E}\{n_1 n_{2D}^{(m)T}\} \) can be evaluated as follows:
\[
\mathcal{E}\{n_1(i) n_{2D}^{(m)}(j)\} = (1 - \delta^{(m)}(i))R_{12}(i, \ell^{(m)}(j))
\]
\[
+ \delta^{(m)}(j)R_{12}(i + 1, \ell^{(m)}(j) + 1) + \delta^{(m)}(j)\delta^{(m)}(j)R_{12}(i + 1, \ell^{(m)}(j) + 1)
\]  
which upon substitution of (14) and (15) in (12) leads to the establishment of the matrix \( \tilde{R}_n^{(m)} \).

Several observations are worth mentioning. First, (14) implies that the noise \( n_{2D}^{(m)}(i) \) is, in general, a nonstationary sequence even when \( n_2(i) \) is a stationary

---

1. Two random (vertical) vectors \( x \) and \( y \) are said to be orthogonal if \( \mathcal{E}\{xy^T\} = 0 \).
random process. Second, (14) indicates that if $n_2(i)$
- is a white sequence, the noise $n_{2D}^{(m)}(i)$ is ordinarily
- a colored sequence; however, if $D^{(m)}(i), \forall m,i,$ is
- approximated to integer numbers ($\delta = 0$), then $n_{2D}^{(m)}(i)$
- may be a sample function of a white sequence as well.\(^2\)
- Finally, (15) suggests that the cross-correlation between
- $n_1(i)$ and $n_{2D}^{(m)}(i)$ is zero if $R_{12}(i,j) = 0$.

C. Estimation of $G$

The elements of the gradient matrix $G$ may be
- measured directly from $r_1(i)$ utilizing an algorithm
- similar to the algorithm (3); that is,

$$g^{(m+1)}(i) = g^{(m)}(i) + \sum_j K_i^{(m+1)}(i,j)[r_1(j) - g^{(m)}(j)]$$

(16)

where $i, j = 1, 2, \ldots, I; m_i = 1, 2, \ldots, M_i$, $g^{(m)}(i)$ is the
- estimate of $s(i)$ at $m$'th iteration, $g^{(0)}(i) = E\{s(i)\}$, and
- $K_i^{(m+1)}(i,j)$ is a weighting function to be determined.
- It follows from (16) that

$$g^{(m+1)} = (1 - K_i^{(m+1)}g^{(m)}) - K_i^{(m+1)}n_1$$

(17)

where $K_i = [K_i(i,j)]; i, j = 1, 2, \ldots, I$, $g^{(m)} = s - g^{(m)}$,
- and $s = [s(1) s(2) \ldots s(I)]^T$. In view of assumption 2
- and the fact that $g^{(0)}(i) = E\{s(i)\}$, it is observed from
- (17) that the estimates $g^{(m)}$, $\forall m_i$, are unbiased.
- The matrix $K_i$ can be evaluated in a manner parallel to that
- of the derivation of $K_D$ (subsection A). Henceforth, assuming that $s$ and $n_1$ are uncorrelated,\(^3\)
- we obtain

$$K_i^{(m+1)} = [\tilde{R}_i^{(m)}]^{-1}$$

(18)

where $\tilde{R}_i^{(m)}$ is the covariance matrix of $g^{(m)}$.
- Correspondingly, we have

$$\tilde{R}_i^{(m+1)} = [1 - K_i^{(m+1)}] \tilde{R}_i^{(m)}$$

(19)

Note that the assumption $g^{(0)}(i) = E\{s(i)\}$ implies that
- $\tilde{R}_i^{(0)} = C_i$, where $C_i = [C_i(i,j)]; i, j = 1, 2, \ldots, I$, is the
- covariance matrix of $s$.
- The entire process of initialization and updating of the estimator is highlighted in Fig. 2.

IV. SIMULATION EXPERIMENTS

This section describes the results of five sets of simulation experiments involving the variable delay estimator (3).
- Experiment I presents the sensitivity of the algorithm to the additive noises $n_1(i)$ and $n_2(i)$.

\(^2\)An apparent violation to this assessment is when $D^{(m)}(i) = -i$, \n- $\forall i, m$.

\(^3\)This assumption is only used for analytical convenience.

Experiment II is focused on the responsiveness of the algorithm to variation of the delay covariance function. In Experiment III, we study the effect of the initial delay error on the acquisition of the algorithm. Experiment IV is devoted to the impact of the data length on the dynamics of the algorithm and in Experiment V, we confine our attention to the accuracy of the estimates by investigating the mean-error function.

The source signal $s(i), i = 1, 2, \ldots, I,$ is assumed to be a sample function from a zero-mean band pass process with the frequency range $1 < \omega < 2$, where $\omega$ is the normalized frequency with $\omega = \pi$ representing half of the sampling frequency. A sample function of this process along with its corresponding covariance function is depicted in Fig. 3.

The delay waveform is assumed to be a sample function from a ramp process defined as

$$D(i) = d_1i + d_2$$

(20)

where $d_1$ and $d_2$ are assumed to be uncorrelated zero-mean random variables\(^4\) with the respective variances $\sigma_{d1}^2 = 1.2$ and $\sigma_{d2}^2 = 2000$. Because the delay $D(i)$ is assumed to be zero mean, it follows from assumption 1 that $D^{(0)}(i) = 0$ and, therefore, from (12) we have

$$\tilde{R}_i^{(0)} = R_{s1} + R_{s2} - R_{12} - R_{21}$$

(21)

\(^4\)The values of $d_1$ and $d_2$ in the experiments are set to $-1$ and $40$, respectively.
where \( R_{n2} = [R_{n2}(i, j)], R_{12} = [R_{12}(i, j)] \), and \( i, j = 1, 2, \ldots, I \). The noise process \( n_1(i) \) is assumed to be a sample function from Markov-1 sequence with the covariance matrix

\[
R_{n1} = \sigma_n^2 \delta[i-j] \tag{22}
\]

where \( \sigma_n^2 \) is the variance and \( \rho \) is correlation coefficient of the sequence. The algorithm for generating the stationary Markov sequence is reported in [11] and is as shown below:

\[
n_1(i) = \rho n_1(i-1) + w(i) \\
n_1(1) = \xi w(1) \tag{23}
\]

where \( i = 2, 3, \ldots, I \),

\[
\xi = \frac{1}{1-\rho^2}; \quad \sigma_n^2 = \frac{1}{1-\rho^2} \sigma_w^2 \tag{24}
\]

and \( \{w(i)\} \) is a white sequence with variance \( \sigma_w^2 \). In all experiments, we assume \( \rho = 0.8 \). The noise \( n_2(i) \) is correlated with \( n_1(i) \) and is generated as

\[
n_2(i) = \alpha \sum_{\ell=1}^{5} n_1(i-\ell), \quad i = 1, 2, \ldots, I \tag{25}
\]

where \( \alpha \) is a normalization factor. Henceforth, the covariance function of \( n_2(i) \) becomes

\[
R_{n2}(i, j) = \sigma_n^2 \sum_{\ell_1=1}^{5} \sum_{\ell_2=1}^{5} R_{n1}(i-\ell_1, j-\ell_2)
\]

\[
= \sigma_n^2 \alpha^2 \sum_{\ell_1=1}^{5} \sum_{\ell_2=1}^{5} \rho^{\lvert \ell_1-j+\ell_2 \rvert} \tag{26}
\]

In all experiments, we assume the variances of \( n_1(i) \) and \( n_2(i) \) are equal; therefore, we set

\[
\alpha = \left[ \sum_{\ell_1=1}^{5} \sum_{\ell_2=1}^{5} \rho^{\lvert \ell_1-j+\ell_2 \rvert} \right]^{-1/2} \tag{27}
\]

Accordingly, the cross-covariance function \( R_{12}(i, j) \) is shown to be

\[
R_{12}(i, j) = \alpha \sum_{\ell=1}^{5} R_{n1}(i, j-\ell) = \sigma_n^2 \alpha \sum_{\ell=1}^{5} \rho^{\lvert i-\ell+j \rvert}. \tag{28}
\]

It should be emphasized that the estimates \( \hat{D}^m(i) \) are, in general, noninteger. Thus, we use the linear interpolation depicted in Fig. 1 to obtain the inter-sample values \( r_2(i+\hat{D}^m(i)) \).

We define the normalized mean-squared error (NMSE) as follows:

\[
\text{NMSE}(m) = \frac{1}{\text{NF}} \sum_i [D(i) - \hat{D}^m(i)]^2 \tag{29}
\]

where \( \text{NF} = \sum_i D^2(i) \) is a normalization factor to let \( \text{NMSE}(0) = 1 \). In Experiments I-IV, the NMSE is depicted and is used as a performance metric of the algorithm. In addition, we define the signal-to-noise ratio (SNR):

\[
\text{SNR} = 10 \log(\sigma_s^2/\sigma_n^2) \tag{30}
\]

where \( \sigma_s^2 \) is the variance of the \( s(i) \) and is set equal to one.

**Experiment I:** This experiment deals with the sensitivity of the algorithm to SNR. We assume that the data length \( I = 80 \). Fig. 4 exhibits the NMSE of the ramp delay (20) for four different scenarios for which SNR is 10, 0, -5, and -10 dB. Clearly, as we decrease SNR, the estimation error increases accordingly.

**Experiment II:** This experiment is focused on the reaction of the algorithm to the variation of the covariance function of the delay vector. We assume that SNR = 0 dB and \( I = 80 \). The variances \( \sigma_{d1}^2 \) and \( \sigma_{d2}^2 \) are increased as depicted in Fig. 5. As seen from this figure, the NMSE of the ramp delay (20) increases as the variances become larger. This experiment reveals that the performance of the algorithm can be deteriorated by the lack of knowledge of the delay covariance matrix.

**Experiment III:** This experiment is concerned with the effect of the initial error on the acquisition
Now consider (\(\ast\)) with \(C_0(t) = 0\):

\[
\dot{r}(t) = c(t) \hat{a}(t) + n(t)
\]

\[
\hat{a}(t) = \int_{-\infty}^{\infty} \frac{\phi_0(c(t)' \hat{a}(t)')}{\phi_0^2(c(t))} \left[ g(t) - g(t') \right] dt'
\]

\[
rg(t) - g(t) = \int_{-\infty}^{\infty} \left[ r(t') - c(t') \hat{a}(t') \right] \phi_0(t, t') dt'
\]

or

\[
r(t) - c(t) \hat{a}(t) = \int_{-\infty}^{\infty} \left[ rg(t') - g(t') \right] \phi_0(t, t') dt'
\]

We shall study the role of these \(g_{\alpha, \beta}\) and properties of the resulting processors.
Properties of optimum processor:

P1. the MAP interval estimate \( \hat{a}(t) \) over \([T_i, T_f]\), where

\[
    r(t) = c(t) a(t) + n(t)
\]

can be obtained by using a linear processor.

**Pf.**

This means

\[
    r(t) \rightarrow h_0(t,u) \rightarrow \hat{a}(t) \quad ; \quad t \in [T_i, T_f].
\]

or

\[
    \hat{a}(t) = \int_{T_i}^{T_f} h_0(t,u) r(u) \, du \quad (3)
\]

First, we multiply \( (1) \) by \( c(t) \) and the result is \( (2) \):

\[
    r(t) = \int_{T_i}^{T_f} \left[ c(t) K_a(t,u) c(u) + K_n(t,u) \right] \left[ r(u) - g(u) \right] \, du \quad (4)
\]

\[
    K_r(t,u) = E\left[ r(t) r(u) \right]
\]
\[ r(x) = \int_{T_i}^{T_f} K_r(x,u) \left[ r_g(u) - g(u) \right] \, du \]  

Multiply both sides of (5) by \( h(t,x) \) and integrate w.r.t \( x \),

\[ \int_{T_i}^{T_f} h(t,x) r(x) \, dx = \int_{T_i}^{T_f} \left[ r_g(u) - g(u) \right] \left\{ \int_{T_i}^{T_f} h(t,x) K_r(x,u) \, dx \right\} \, du \]

(1) \implies \text{if } \text{This is equal to } \quad \text{then the left side becomes (3)}

\[ K_{a}(t,u) C(u) = \int_{T_i}^{T_f} h_{o}(t,x) K_{r}(x,u) \, dx \]

(6)

\[ T_i \leq t, x \leq T_f \]

The result of this gives the \( q_{o}^{*} \) for the optimum impulse response \( h_{o}(t,x) \).
\[ R(t) = C(t) a(t) + n(t) \]
\[ r_c(t) + w(t) \]

\[ K_r(x, u) = C(x) K_a(x, u) C(u) + K_c(x, u) + \frac{N_0}{2} \delta(x-u) \]

\[ (7) \quad i \quad (6) = 1 \]

\[ K_a(t, u) C(u) = \int_{T_i}^{T_f} h_0(t, x) \left[ \right] \, dx \]

\[ = \int_{T_i}^{T_f} h_0(t, x) \left[ C(x) K_a(x, u) C(u) + K_c(x, u) + \frac{N_0}{2} h_0(t, u) \right] \, dx \]

In Chapter 4 we obtained that \( n(t) \) is real.
P2 \[ \text{the MAP interval estimate } \hat{a}(t) \text{ over } [T_i, T_f], \]

where

\[ r(t) = c(t) a(t) + n(t) \]

is also the MMSE interval estimate.

**Proof:** Note that for linear modulation, the MAP estimate is efficient, i.e.,

\[ \frac{\partial \ln P(r | A)}{\partial A} = \left[ \hat{a}(R) - A \right] K(A) \]

\[ \frac{\partial A}{\partial A} \]

The CR bound is satisfied with equality

\[ \text{Var} \left[ \hat{a}(R) - A \right] = \text{minimum} \]

Thus, P2 is proved.
Now consider, 

\[ \text{Desired Linear operation} \]

\[ K_f(t,v) \rightarrow d(t) \]

\[ x(u) \]

\[ y(u) + n(u) \]

\[ r(u) \]

\[ u \in [T_i, T_f] \]

\[ q(u) \]

\[ k_f(u,v) \]

\[ \text{Linear operation} \]

\[ c(u) \]

\[ \text{Aside:} \]

\[ s(u-v) \rightarrow h(u,v) \rightarrow y(u) \]

\[ s(u-v) \rightarrow h(u,v) \]

We wish to operate on \( r(u) \), \( u \in [T_i, T_f] \) to obtain an estimate of \( d(t) \). In other words, we wish to find \( \hat{d}(t) \) such that

\[ J_p(t) = E \frac{1}{2} \int [d(t) - \hat{d}(t)]^2 e(t) \]

is minimized.
P3. The MMSE linear estimate is the output of a linear processor as defined:

\[ r(t) \xrightarrow{h_0(t, \tau)} \hat{d}(t) \]

where \( h_0(t, \tau) \) is a real-\( \tau \) - the integral eye:

\[ K_{dr}(t, u) = \int_{T_i}^{T_f} h_0(t, \tau) K_r(\tau, u) d\tau, \quad T_i < u < T_f \]

**Pf.** Note that

\[ \hat{d}(t) = \int_{T_i}^{T_f} h(t, \tau) r(\tau) d\tau \]

Using the orthogonality principle, we have

\[ e_{\hat{d}}(t) = 0, \quad E_{\theta} e_{\hat{d}}(t) r(u)^2 = 0. \]

\[ \forall t, u. \]
\[ E \mathbb{E} \left[ d(t) - \hat{d}(t) \right] r(u) = 0. \]

\[ E \left\{ d(t) r(u) \right\} - E \left\{ \hat{d}(t) r(u) \right\} = 0 \]

\[ K_{d,r}(t,u) = \int_{T_i}^{T_f} h(t, \tau) K_r(\tau, u) d\tau, \quad T_i < u < T_f \]

which is the desired result.

**Remark** We emphasize that we have not used the Gaussian assumption. Later, we show that a linear processor is the best of all possible processors for the MSE criterion when Gaussian assumption is invoked.