A submission for IEEE International Symposium on Information Theory, ISIT 2003.

Capacity of a Multi Output Channel with Distributed Processing

Natanael Peranginangin   Muriel Médard   Robert G. Gallager

Abstract**—***To help understand distributed processing in the context of channel coding, we study distributed processing in a single input multi output (SIMO) channel with memory. Distributed processing in the context of source coding has been studied. One problem which arises in the context of source coding is the CEO problem [2], [9], [8], [4]. A parallel between our study and [4] is that distributed processing is successively structured. Rather than consider distortion metrics, we consider transmission capacity. We develop a one dimensional (1D) Kalman filter for a SIMO channel with no intersymbol interference (ISI). By one dimensional, we mean that estimation by Kalman filtering proceed from the first receiver to the next and successively to the last receiver at every time step. We show that the capacity of the channel with centralized processing is the same as that of the channel with distributed processing by 1D Kalman filtering. We extend the channel model to a channel with ISI. ISI entails the problem of infinite memory and delay at each receiver if distributed processing is done by means of 1D Kalman filtering, rendering successive estimation by 1D Kalman filter to be impractical. To mitigate the problem of infinite memory and delay, we develop a two dimensional (2D) Kalman filter. By two dimensional, we mean that estimation by Kalman filtering proceeds from the first receiver to the next and successively to the last receiver at every time step. Within the same time step, however, the last receiver feedback its estimate to the first receiver, allowing the estimate of the next time step to be based on the present time step. Finally we find the expression for the capacity of the channel with centralized processing and show that it is the same as that of the channel with distributed processing.*

I. Introduction

The objective of this paper is to help understand distributed processing of information in the context of channel coding. We assume a multi output channel subject to both additive white Gaussian noise (AWGN) and intersymbol interference (ISI). We also assume that both the transmitter and the receiver have complete knowledge of the channel. In essence we address two primary problems, namely, whether distributed processing of several output streams in a multi output channel yields the same capacity as that of centralized processing and how distributed processing can be done efficiently at each receiver in the case of channel with ISI.

In general, the distributed processing problem in this paper falls into the class of problems involving the observing of events in space-time and making estimates or predictions based on the phenomena observed. Often the sensor-estimator system is distributed in the sense that data are collected at several spatially separated sites and have to be transmitted to a central estimator/decision maker over communication channels of limited capacity. A problem which belongs to this class and has been studied in the context of source coding is the *CEO* problem. [2] introduced and studied the problem for discrete memoryless source. A contrast based on rate-distortion or distortion-rate function is made between the case when the estimators can convene for the purpose of smoothing the corrupted observation and the case when the estimators can not. [9], [8] extended the study to the special case of continuous Gaussian source and observation. More recently, [4] developed successive encoding strategies for the CEO problem based on generalized Wyner-Ziv encoding. By successive encoding we mean estimators (or 'agents') are ordered and communicate −one to the next− over rate-constrained links, the final agent in the chain being termed as the CEO. Two of the differences between distributed processing in this paper and the successively structured encoding for the CEO problem in [4] are as follows. Distributed processing is performed successively in the context of channel coding with channel capacity being the metric to contrast the case when several receivers process their observed output in a successive and distributed manner with the case when the receivers process their observed output in a centralized manner. Moreover, in the case of single input and multi output (SIMO) channel that we consider throughout, distributed processing is performed in between coding and encoding processes, implying one transmitter ('agent') encodes and only one receiver ('CEO') decodes.

One scenario in which distributed processing applies is the processing of wireless signals to preserve wireline network [7]. The problem addressed by [7] is an architectural problem with respect to the transmission of information between the wireless and wireline domain, namely, the communication architecture that meets the optimal tradeoff between attaining wireless rate gain (in a wireless system with multiple receivers) and minimizing wireline processing thus wireline bandwidth for the desired wireless rate gain.

In section 2 we develop a one dimensional (1D) Kalman filter as means of successively estimating the input sequence of a single input multiple output (SIMO) and AWGN channel with no ISI. One dimensional recursive estimation from one receiver to the next by the Kalman filter yields the same final estimates in contrast with centralized estimation (processing) when the complete set of observed output sequences are fused and processed in a centralized fashion by either a linear least square (LLS) or maximum likelihood (ML) or

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: {nael,medard,gallager})@mit.edu

Bayes' Least Square (BLS) estimator. We find the expression for the capacity of the channel with centralized processing and show that it is the same as that of the channel with distributed processing, i.e., 1D Kalman filtering. In section 3 we extend the channel model of section 2 to a SIMO and AWGN channel with ISI. ISI entails the problem of infinite memory and delay at each receiver if distributed processing is done by means of the 1D Kalman filtering. To mitigate the problem of infinite memory and delay, the channel with distributed processing performs two dimensional (2D) Kalman filtering. By two dimensional, we mean that estimation by Kalman filtering proceeds from the first receiver to the next and successively to the last receiver at every time step. Within the same time step, however, the last receiver feedback its estimate to the first receiver, allowing the estimate of the next time step to be based on the present time step. The final estimates attained by the 2D Kalman filter is the same as either the estimates computed by the impractical 1D Kalman filter or the much more impractical centralized estimator observing the complete set of output sequences and producing an estimates based on linear least square (LLS) or maximum likelihood (ML) or Bayes' least square (BLS) estimation. Finally we find the expression for the capacity of the channel with centralized processing and show that it is the same as that of the channel with distributed processing.

Aiming at preliminary insights to the question of whether distributed processing yields the capacity equal to that of centralized processing in the case of channel with intersymbol interference (ISI), we begin with a model of the channel, i.e., AWGN channel with single input multiple output (SIMO), no intersymbol interference, and both transmitter and receiver have perfect knowledge of the channel.

II. Channel Model: AWGN, SIMO, No ISI, and Perfect Knowledge of Channel at Transmitter and Receiver

A user transmits a symbol X with average energy constraint $E[X^2] \leq E$. There are n receivers, and the channel between the single user and receiver i, denoted as channel i, is corrupted by AWGN noise N_i with average energy σ_i^2. The relation between input symbol X and the output symbol at receiver i, Y_i, is given by

$$Y_i = X + N_i \qquad x = 1, \ldots, n \tag{1}$$

where

X a random variable with zero mean and variance E;

Y_i a random variable with zero mean and variance $(E + \sigma_i^2)$;

N_i $\sim$ Normal $(0, \sigma_i^2)$ for $i = 1, \ldots, n$ and N_i is independent of X and N_j, for $j \in (1, 2, \ldots, n) \setminus i$.

Considering the far smaller order of magnitude in the error rate for current optical links with respect to that of wireless links, we assume the wireline network to be noiseless. In such a noiseless wireline environment, two options arise as to how to process the Y_i's, with each option having a different bearing on bandwidth usage in the wireline domain. In what follows we describe each option.

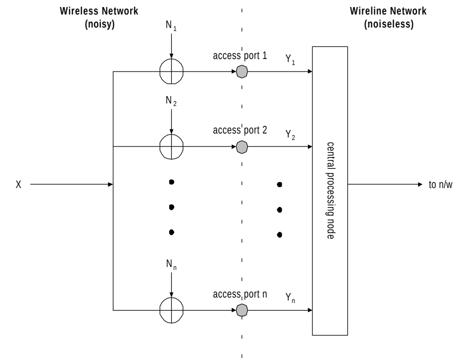

Fig. 1. Channel with centralized processing

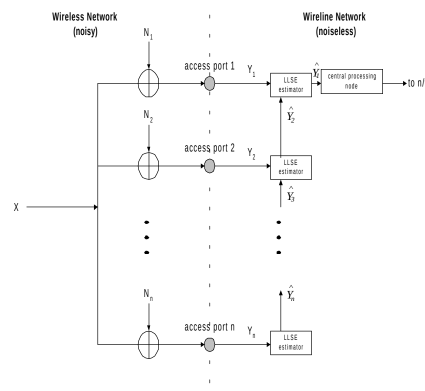

Fig. 2. Channel with distributed processing.

Option 1 : Centralized Processing
As Figure 1 illustrates, at each time step (since we take a discrete time approach), access port i, $i = 1, \ldots, n$, simultaneously sends its received signal, Y_i, $i = 1, \ldots, n$, to the central processing node through the wireline network.

Option 2 : Distributed Processing
As Figure 2 illustrates, this option allows estimation to be done stage by stage at each access port. In particular, access port 1 receives Y_1, then produces a linear least square (LLS) estimate of X, $\hat{Y}_1$, and sends the estimate to access port 2. Access port 2 then processes $\hat{Y}_1$ along with Y_2, producing yet another LLS estimate $\hat{Y}_2$, and send the corresponding estimate to the subsequent access port, i.e., access port 3. In the same fashion, LLS estimation process carries on until the central node receives an estimate, $\hat{Y}_n$, from access port n.

A. One Dimensional (1D) Kalman Filtering Algorithm

Since at any time unit it is possible to have a record of the measurements $\{Y_i\}_{i=1}^n$, distributed processing can be rephrased to be the on-line estimation of the state variable of port n, i.e., $U_n = X$, from the measurements $\{Y_i\}_{i=1}^n$

One such algorithm that yields an on-line unbiased linear least square estimate of the state U_n is the 1D-Kalman filter [1] [10]. The estimate is unbiased in the sense that

$$E[U_n - \hat{U}_n] = 0$$

where $E[\bullet]$ denotes expectation, and $\hat{U}_n = \hat{X}$ is the estimate of $U_n = X$ from $\{Y_i\}_{i=1}^n$, i.e., available measurements. The minimum error variance characteristic simply means that the quantity

$$E[(\hat{U}_n - U_n)^2]$$

can be minimized from the requirement that the estimate be the result of a linear operation on the available measurements [1] [10].

A particularly convenient form for the 1D-Kalman estimation algorithm can be developed in a recursive manner [1] [10], with a state-space model defined by state and output equations

$$U_{i+1} = U_i \quad (2)$$

$$Y_{i+1} = U_{i+1} + N_{i+1} \quad (3)$$

Note that the output equation (3) follows from (1) and (2).

Let us define

- $\hat{U}[i|k]$ linear least-square estimate of U_i, based on observations from port 1 to port $k (k \leq i)$;
- K_i port varying Kalman gain (a scalar);
- $\lambda_e[i|k]$ error covariance matrix, based on observations from port 1 to port $k (k \leq i)$, i.e., $E\left[(U_i - \hat{U}[i|k])^2\right]$;
- σ_i^2 measurement noise variance at port i, i.e., $E[(N_i)^2]$;

The on-line 1D Kalman filtering algorithm is as follows.

1) Initialize the prediction and its associated error variance according to

$$\hat{U}[1|0] = 0$$

$$\lambda_e[1|0] = E$$

and let $i = 1$.

2) If $i \leq n$ go to the next step, if $i > n$ then send $\hat{Y}_n$ to central processing node and end process.

3) Port i computes the Kalman gain matrix

$$K_i = \lambda_e[i|i-1]\frac{1}{(\lambda_e[i|i-1] + \sigma_i^2)}$$

and generate the filtered estimate and its associated error covariance from the corresponding prediction quantities according to

$$\hat{U}[i|i] = \hat{U}[i|i-1] + K_i\left(Y_i - \hat{U}[i|i-1]\right)$$

$$\lambda_e[i|i] = \lambda_e[i|i-1] - K_i\lambda_e[i|i-1]$$

4) Port i generates the next prediction and its associated error covariance from the corresponding filtered quantities according to

$$\hat{U}[i+1|i] = \hat{U}[i|i]$$

$$\lambda_e[i+1|i] = \lambda_e[i|i]$$

5) Send $\hat{U}[i+1|i]$ and $\Lambda_e[i+1|i]$ to port $i+1$.

6) Increment i and go to step 2.

The following theorem follows immediately.

Theorem II.1: Let $\hat{X}$ be the linear least square estimate of X based on observing $\{Y_i\}_{i=1}^n$ and $\hat{Y}_n$ be an estimate attained by port n under the distributed processing scheme. Then

$$\hat{X} = \hat{Y}_n$$

B. Capacity of the Channel with Distributed Vs. Centralized Processing

Consider the channel with *centralized* processing, i.e, a channel whose input X, noise $\bar{N} = \{N_i\}_{i=1}^n$, and output $\bar{Y} = \{Y_i\}_{i=1}^n$ are related by (1), with symbol energy constraint $E[X^2] \leq E$. Moreover, let us define a new channel which is a twist from the *centralized* channel. Instead of having a discrete time channel with input, X, and output $\bar{Y}$, i.e., a channel whose input-output relation is that of the channel with *centralized* processing defined by (1), we reduce the channel to one which has input, $\bar{X}$, and output, $\hat{\bar{X}}$, where $\hat{X}$ is the Linear Least Square (LLS) estimate of X, that is,

$$\hat{\bar{X}} = \mathbf{A}\bar{Y} \quad (4)$$

where $\mathbf{A}$ is a matrix with the appropriate dimension such that $E\left[(X - \hat{X})^2\right]$ is minimized. Let us name the channel with input and output relation in (4) as the channel with *distributed* processing.

Lemma II.2:

$$\sup_{\bar{X}:E[X^2]\leq E} I(\bar{Y};X) = \frac{1}{2}\log\left(1 + \frac{E}{\frac{\Pi_{k=1}^n \sigma_k^2}{\sum_{i=1}^n \Pi_{k\in(1,2,\ldots,n)\setminus i}\sigma_k^2}}\right)$$

Proof. Constrained by the number of pages, we exclude the proof. □

Lemma II.3: Let C_c be the capacity of the channel under centralized processing (in bits per channel input symbol when logarithm is taken to the base 2), then

$$C_c = \frac{1}{2}\log\left(1 + \frac{E}{\frac{\Pi_{k=1}^n \sigma_k^2}{\sum_{i=1}^n \Pi_{k\in(1,2,\ldots,n)\setminus i}\sigma_k^2}}\right)$$

Proof. Taking into account the difference between the SISO AWGN and the SIMO AWGN (*centralized* channel), i.e., the output of the SIMO channel is a vector, the proof is similar to the standard proof for SISO AWGN channel, [3, pages 244–245]. □

Lemma II.4:

$$\sup_{X:E[X^2]\leq E} I(X;\bar{Y}) = \sup_{X:E[X^2]\leq E} I(X;\hat{X})$$

Proof. Constrained by the number of pages, we exclude the proof. □

Theorem II.5: Let C_d be the capacity of the channel under distributed processing (in bits per channel input symbol when logarithm is taken to the base 2), then

$$C_d = C_c = \frac{1}{2}\log\left(1 + \frac{E}{\frac{\Pi_{k=1}^n \sigma_k^2}{\sum_{i=1}^n \Pi_{k\in(1,2,\ldots,n)\setminus i}\sigma_k^2}}\right)$$

Proof. ($\Rightarrow$) Recall Theorem II.1, Lemma II.2, Lemma II.4. Taking into account the difference between the channel under centralized processing and the channel under distributed processing, i.e., the output of the channel under distributed processing is a LLS estimate of $\{Y_i\}_{i=1}^n$, the proof is similar to the forward proof of Lemma II.3.

($\Leftarrow$) Constrained by the number of pages, we exclude the proof of the converse. □

III. Channel Model: AWGN, SIMO, ISI, and Perfect Knowledge of Channel at Transmitter and Receiver

We now consider the single input multiple output (SIMO) case when there is inter symbol interference (ISI) at each channel i. A user sends a sequence of symbols, $\bar{X} =$

$\{X_j\}_{j=1}^{\infty}$. The input process is corrupted by a white Gaussian noise process which are independent for the n receivers. The relation between the input and output process of the channel is

$$Y_{i_j} = \sum_{k=0}^{M_i} h_{i_k} X_{j-k} + N_{i_j} \quad i = 1, \ldots, n \quad j = 1, \ldots, \infty \tag{5}$$

where

$\bar{Y}$ be the column vector of output process $\{\{Y_{i_j}\}_{j=1}^{\infty}\}_{i=1}^{n}$.

$\bar{X}$ be the column vector of the zero mean input process $\{X_j\}_{j=1}^{\infty}$ satisfying average symbol energy constraint $E[{X_i}^2] \leq E, \ i = 1, \ldots, \infty;$

$\{h_{i_k}\}_{k=0}^{M_i}$ the finite impulse response (FIR) (with memory M_i) of the filter corresponding to access port i;

$\{N_{i_j}\}_{j=1}^{\infty}$ a white Gaussian noise process with mean zero and average energy $E[{N_{i_j}}^2] = \sigma^2$ for all i, j.

We shall refer to the channel whose input-output relationship is described by (5) as the channel with *centralized* processing.

Moreover, let us define a new channel which is a twist from the channel with *centralized* processing. Instead of having a discrete time channel with input, $\bar{X}$, and output $\bar{Y}$, i.e., a channel whose input-output relation is that of the channel with *centralized* processing defined by (5), we reduce the channel to one which has input, $\bar{X}$, and output, $\hat{\bar{X}}$, where $\hat{\bar{X}}$ is the Linear Least Square (LLS) estimates of $\bar{X}$, that is,

$$\hat{\bar{X}} = \mathbf{A}\bar{Y} \tag{6}$$

where $\mathbf{A}$ is a matrix with the appropriate dimension such that$E\left[(\bar{X} - \hat{\bar{X}})^T(\bar{X} - \hat{\bar{X}})\right]$ is minimized. Let us name the channel with input and output relation in (6) as the channel with *distributed* processing .

A. Distributed Processing with 2D-Kalman Filter

As figure 2 shows, a central problem with the channel under distributed processing is as follows. For each access port to compute distributed and bottom-up linear least square estimates of the input process $\{X_j\}_{j=1}^{\infty}$, the receiver at access port i ($i = 1, \ldots, n$) is forced to store an infinite sequence of the observed output process $\{Y_{i_j}\}_{j=1}^{\infty}$. With an infinite sequence of observed output process being stored at each access port, one dimensional port-by-port Kalman filtering is then performed from the bottom port to the top one, thus attaining the final estimates, i.e., $\{\hat{X}_j\}_{j=1}^{\infty}$. The implications are twofolds, i.e., infinite memory and delay. Each port has to have infinite memory for storing the infinite output sequence it receives. Moreover, there is an infinite delay before the estimates of $\{X_j\}_{j=1}^{\infty}$, i.e., $\{\hat{X}_j\}_{j=1}^{\infty}$ can finally be computed at the central processing node.

To alleviate the problem of infinite memory and delay, the channel under distributed processing performs two dimensional Kalman filtering as a mean for computing linear least square estimates of the input process. Instead of storing an infinite sequence of the output process at each access port, access port 1 performs discrete time Kalman filtering, thus allowing recursive estimation of the input process at every time unit. With some finite memory at each access port, Kalman filtering is again performed port-by-port from the bottom port to the top one, i.e., port 1 to port n respectively, yielding the final estimates of the input process, i.e., $\{\hat{X}_j\}_{j=1}^{\infty}$. Hence the channel under distributed processing performs a combining of time-wise and port-by-port Kalman filtering, i.e., a two dimensional filtering operation.

We will show that two dimensional Kalman filtering yields the same linear least square estimates of the input process as that which is resulted from the one dimensional Kalman filtering when an infinite sequence of observed output process is in store at each access port.

B. A Base Model for Estimation

As (5) suggests, the base model chosen to represent the relation between input and observed output of the channel corresponding to access port i ($i = 1, \ldots, n$) is

$$Y_{i_j} = \sum_{k=0}^{M} h_{i_k} X_{j-k} + N_{i_j} \tag{7}$$

where

$\{X_j\}_{j=1}^{\infty}$ a white wide sense stationary (WSS) input process with zero mean and unit variance;

$\{h_{i_k}\}_{k=0}^{M_i}$ the finite impulse response (FIR), with memory M_i, of the filter corresponding to access port i;

M be $\max_i M_i$;

$\{N_{i_j}\}_{j=1}^{\infty}$ a white Gaussian noise process with mean zero and average energy $E[{N_{i_j}}^2] = \sigma^2$ for all i, j.

Moreover it is assumed that $X_{-M} = \ldots = X_0 = 0$.

Denote $\bar{U}_{i_j}$ to be the vector of states at time j for the channel associated to access port i. Then (7) suggests that the dynamics of the input process can be represented by the state equation

$$\bar{U}_{i_{j+1}} = \mathbf{F}\bar{U}_{i_j} + \mathbf{G}X_{j+1} \tag{8}$$

$$\bar{U}_{(i+1)_j} = \bar{U}_{i_j} \tag{9}$$

where

$$\begin{aligned} \bar{U}_{i_{j+1}} &= \begin{bmatrix} X_{j+1} & X_j & \cdots & X_{j-M+1} \end{bmatrix}^T \\ \bar{U}_{i_j} &= \begin{bmatrix} X_j & X_{j-1} & \cdots & X_{j-M} \end{bmatrix}^T \\ \bar{U}_{(i+1)_j} &= \begin{bmatrix} X_j & X_{j-1} & \cdots & X_{j-M} \end{bmatrix}^T \end{aligned}$$

(all are column vectors with $(M+1)$ components)

and

$$\mathbf{F} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}, \mathbf{G} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Relating (7) to (8) and (9), the channel output at time j can be described in term of the state vector at time j

$$Y_{i_j} = \mathbf{C_i}\bar{U}_{i_j} + N_{i_j} \tag{10}$$

where $\mathbf{C_i}$ is an 1 X $(M+1)$ matrix, i.e., $\mathbf{C_i} = \begin{bmatrix} h_{i_0} & h_{i_1} & \cdots & h_{i_M} \end{bmatrix}$

The estimation problem is now stated: given the foregoing channel model and the statistics of the input signal and the measurement noise, it is desired to obtain an on-line estimation procedure that yields a linear least square estimates of the input sequence $\{X_j\}_{j=1}^{j=\infty}$ at some delayed time $(j+M)$.

2D-Kalman Filtering Algorithm: Since at any time j it is possible to have a record of the measurements $\{Y_{i_1}\}_{i=1}^{n}$, $\{Y_{i_2}\}_{i=1}^{n}$, ..., $\{Y_{i_j}\}_{i=1}^{n}$, the preceding estimation problem can be rephrased to be the on-line estimation of the $(M+1)$ components of the state vector of port n at time j, i.e., $\bar{U}_{n_j} = \begin{bmatrix} X_j & X_{j-1} & \cdots & X_{j-M} \end{bmatrix}^T$, from the measurements $\{Y_{i_1}\}_{i=1}^{n}$, $\{Y_{i_2}\}_{i=1}^{n}$, ..., $\{Y_{i_j}\}_{i=1}^{n}$.

One such algorithm that yields an on-line unbiased linear least square estimate of the complete state vector $\bar{U}_{n_j}$ is the 2D-Kalman filter [1] [10]. The estimate is unbiased in the sense that

$$E[\bar{U}_{n_j} - \hat{\bar{U}}_{n_j}] = 0$$

where $E[\bullet]$ denotes expectation, and $\hat{\bar{U}}_{n_j} = \begin{bmatrix} \hat{X}_j & \hat{X}_{j-1} & \cdots & \hat{X}_{j-M} \end{bmatrix}^T$ is the estimate of $\bar{U}_{n_j} = \begin{bmatrix} X_j & X_{j-1} & \cdots & X_{j-M} \end{bmatrix}^T$ from $\{Y_{i_1}\}_{i=1}^{n}$, $\{Y_{i_2}\}_{i=1}^{n}$, ..., $\{Y_{i_j}\}_{i=1}^{n}$, i.e., available measurements at time j. The minimum error variance characteristic simply means that the quantity

$$E[(\bar{U}_{n_j} - \hat{\bar{U}}_{n_j})^T(\bar{U}_{n_j} - \hat{\bar{U}}_{n_j})]$$

can be minimized from the requirement that the estimate be the result of a linear operation on the available measurements [1] [10].

A particularly convenient form for the 2D-Kalman estimation algorithm can be developed in a recursive manner [1] [10]. More precisely, for the estimation model defined by (8), (9), and (10), let us define

- $\hat{\bar{U}}_{i|t}[j|k]$ linear least-square estimate of $\bar{U}_{i_j}$, based on observations from port 1 to port $t (t \le i)$ which spans from time 1 to time $k (k \le j)$, i.e., $\{\{Y_{p\ q}\}_{p=1}^{l}\}_{q=1}^{k}$;
- $\bar{K}_{i_j}$ time and port varying Kalman gain (an $(M+1)$ X 1 vector);
- $\mathbf{\Lambda_{e_{i|t}}}[j|k]$ error covariance matrix, based on observations from port 1 to port $t (t \le i)$ which spans from time 1 to time $k (k \le j)$, $\{\{Y_{p\ q}\}_{p=1}^{l}\}_{q=1}^{k}$, i.e., $E\left[(\bar{U}_{i_j} - \hat{\bar{U}}_{i|t}[j|k])(\bar{U}_{i_j} - \hat{\bar{U}}_{i|t}[j|k])^T\right]$;
- σ^2 measurement noise covariance = $E[(N_{i_j})^2]$ for all i,j;
- $\mathbf{I}, \mathbf{0}$ the $(M+1)$ X $(M+1)$ identity matrix and the column vector with all its $(M+1)$ components being 0, respectively.

The algorithm is as follows.

1) Initialize the prediction and its associated error variance according to

$$\begin{aligned} \hat{\bar{U}}_{1|1}[1|0] &= \mathbf{0} \\ \mathbf{\Lambda_{e_{1|1}}}[1|0] &= \mathbf{I} \end{aligned}$$

and let $j = 1$.

2) Let $i = 1$

3) Port 1 computes the Kalman gain matrix

$$\bar{K}_{1_j} = \mathbf{\Lambda_{e_{1|1}}}[j|j-1]\mathbf{C_1}^T \left(\mathbf{C_i}\mathbf{\Lambda_{e_{1|1}}}[j|j-1]\mathbf{C_1}^T + \sigma^2\right)$$

and generate the filtered estimate and its associated error covariance from the corresponding prediction quantities according to

$$\begin{aligned} \hat{\bar{U}}_{1|1}[j|j] &= \hat{\bar{U}}_{1|1}[j|j-1] + \bar{K}_{1_j}\left(Y_{1_j} - \mathbf{C_1}\hat{\bar{U}}_{1|1}[j|j-1]\right) \\ \mathbf{\Lambda_{e_{1|1}}}[j|j] &= \mathbf{\Lambda_{e_{1|1}}}[j|j-1] - \bar{K}_{1_j}\mathbf{C_1}\mathbf{\Lambda_{e_{1|1}}}[j|j-1] \end{aligned}$$

4) While $i \le n$ perform as follows. If $i > n$ go to step 5.
   a) Port i generates the next prediction and its associated error covariance from the corresponding filtered quantities according to

$$\begin{aligned} \hat{\bar{U}}_{i+1|i}[j|j] &= \hat{\bar{U}}_{i|i}[j|j] \\ \mathbf{\Lambda_{e_{i+1|i}}}[j|j] &= \mathbf{\Lambda_{e_{i|i}}}[j|j] \end{aligned}$$

   b) Send $\hat{\bar{U}}_{i+1|i}[j|j]$ and $\mathbf{\Lambda_{e_{i+1|i}}}[j|j]$ to port $i+1$.
   c) Increment i.
   d) Port i computes the Kalman gain matrix

$$\bar{K}_{i_j} = \mathbf{\Lambda_{e_{i|i-1}}}[j|j]\mathbf{C_i}^T\left(\mathbf{C_i}\mathbf{\Lambda_{e_{i|i-1}}}[j|j]\mathbf{C_i}^T + \sigma^2\right)^{-1}$$

   and generate the filtered estimate and its associated error covariance from the corresponding prediction quantities according to

$$\begin{aligned} \hat{\bar{U}}_{i|i}[j|j] &= \hat{\bar{U}}_{i|i-1}[j|j] + \bar{K}_{i_j}\left(Y_{i_j} - \mathbf{C_i}\hat{\bar{U}}_{i|i-1}[j|j]\right) \\ \mathbf{\Lambda_{e_{i|i}}}[j|j] &= \mathbf{\Lambda_{e_{i|i-1}}}[j|j-1] - \bar{K}_{i_j}\mathbf{C_i}\mathbf{\Lambda_{e_{i|i-1}}}[j|j-1] \end{aligned}$$

   e) Go to step 4.

5) Port n generates the next prediction and its associated error covariance from the corresponding filtered quantities according to

$$\begin{aligned} \hat{\bar{U}}_{1|1}[j+1|j] &= \mathbf{F}\hat{\bar{U}}_{n|n}[j|j] \\ \mathbf{\Lambda_{e_{1|1}}}[j+1|j] &= \mathbf{F}\mathbf{\Lambda_{e_{i|i}}}[j|j]\mathbf{F}^T + \mathbf{G}\mathbf{G}^T \end{aligned}$$

6) Send $\hat{\bar{U}}_{1|1}[j+1|j]$ and $\mathbf{\Lambda_{e_{1|1}}}[j+1|j]$ to port 1.

7) Increment j and go to step 2.

The following theorem follows immediately.

Theorem III.1: Let $\hat{\bar{X}}$ be the linear least square estimate of X based on observing $\{\{Y_{i_j}\}_{j=1}^{\infty}\}_{i=1}^{n}$ and $\bar{U}_{n_j}^{M+1}$ be the $(M+1)$-th component of the state vector estimate attained by port n at time j, i.e., $\bar{U}_{n_j}$, under the distributed processing scheme. Then

$$\hat{\bar{X}} = \{\bar{U}_{n_j}^{M+1}\}_{j=M+1}^{\infty}$$

Lemma III.2: For any colored wide sense stationary process (WSS) $\{V_j\}_{j=1}^{\infty}$, there exists a causal filter with memory H and impulse response $\{g_0, g_1, \ldots, g_H\}$. Moreover

$$V_j = \sum_{k=0}^{H} g_{i_k} X_{j-k}$$

and $\{X_j\}_{j=1}^{\infty}$ is a white WSS process with $E\left[(X_j)^2\right] = 1,\ j = 1, \ldots, \infty$.

Proof. From [10] or [1], spectral factorization, i.e., Gram-Schmidt orthogonalization, will yield a causal filter which satisfies the statement of the theorem. □

Recall (7). The input sequence is the white WSS process,

$\{X_j\}_{j=1}^{\infty}$. By lemma III.2, we can perform 2D Kalman filtering on the colored WSS process $\{V_j\}_{j=1}^{\infty}$ by replacing $\{\{h_{i_j}\}_{i=1}^{n}\}_{j=0}^{M}$ with $\{\{\tilde{h}_{i_j}\}_{i=1}^{n}\}_{j=0}^{M+H}$ where

$$\tilde{h}_{i_j} = \sum_{k=0}^{\min\{M,H\}} g_{i_k} h_{i_{j-k}} \ , \quad j = 0, \ldots, (M+H)$$

and replace M with $(M+H)$. In the case where $H = \infty$, a truncation strategy would be required to find the most sensibly finite amount of memory, $\tilde{H}$, to assign in the state vector of the state-space estimation model so that the error covariance gap between the two estimates, i.e., $\hat{\bar{X}}$ and $\{\bar{U}_{n_j}^{M+\tilde{H}+1}\}_{j=M+\tilde{H}+1}^{\infty}$ (the sequence of the $(M+\tilde{H}+1)$-th component of the state vector estimate) is reasonably small. This problem is not elaborated in this paper and is a subject for further study.

C. Capacity of Distributed Vs. Centralized Processing

We now show that capacity of channel under distributed processing, C_d, is the same as that of centralized processing, C_c.

Theorem III.3: Let C_c be the capacity of the channel under centralized processing (in bits per channel input symbol when logarithms are taken to the base 2). Then

$$C_c = (2\pi)^{-1} \int_0^{\pi} \log \left[\max\left(\Theta \frac{\Pi_{i=1}^{n} |H_i(\lambda)|^{-2}}{\sum_{k=1}^{n} \Pi_{i\in(1,2,\ldots,n)\setminus i} |H_i(\lambda)|^{-2}} \ , 1 \right) \right] d\lambda$$

where $H_i(\lambda)$ is the channel transfer function given by

$$H_i(\lambda) = \sum_{j=0}^{M_i} H_{i_j} \exp^{-lj\lambda} \ , \quad l = \sqrt{-1}$$

(periodic in λ with period 2π) and where the parameter θ is the solution of

$$\int_{0 (H_i(\lambda)\neq 0 \ \forall i)}^{\pi} \max\left(\Theta - \frac{\Pi_{i=1}^{n} |H_i(\lambda)|^{-2}}{\sum_{k=1}^{n} \Pi_{i\in(1,2,\ldots,n)\setminus i} |H_i(\lambda)|^{-2}} \ , 0 \right) d\lambda = \frac{\pi E}{\sigma^2}$$

Moreover, the capacity-achieving q_N, the inputs $\{X_j\}_{j=\infty}^{\infty}$, are correlated Gaussian random variables with mean zero and covariances r_n, $-\infty \leq n \leq \infty$, given by

$$r_n = E\left[X_{k+n} X_k\right] = (\pi)^{-1} \int_0^{\pi} S_X(\lambda) \cos(n\lambda) d\lambda$$

where the input power spectral density satisfies

$$S_X(\lambda) = \begin{cases} \sigma^2(\theta - K(\lambda)^{-2}), & \theta K(\lambda)^2 > 1 \quad |\lambda| \leq \pi, \\ 0, & otherwise \end{cases}$$

with $$K(\lambda) = \frac{\Pi_{i=1}^{n} |H_i(\lambda)|^{-2}}{\sum_{k=1}^{n} \Pi_{i\in(1,2,\ldots,n)\setminus i} |H_i(\lambda)|^{-2}}$$

In particular, capacity is achieved when all inputs X_j, $-\infty \leq j \leq \infty$, have the same average energy $E[X_j^2] = r_0 = E$.

Proof. The proof is an extension to the proof for the capacity of the single input and single output (SISO) discrete-time Gaussian channel with intersymbol interference [6]. Constrained by the number of pages, we exclude the rest of the proof. □

Lemma III.4:

$$\sup_{\bar{X}: E[X_i{}^2] \leq E, \forall i} I(\bar{X}; \bar{Y}) = \sup_{\bar{X}: E[X_i{}^2] \leq E, \forall i} I(\bar{X}; \hat{\bar{X}})$$

Proof. Constrained by the number of pages, we exclude the proof. □

Theorem III.5: The capacity of channel with distributed processing, C_d, is equal to the capacity of channel with centralized processing, C_c.

Proof. Recall theorem III.1, theorem III.3 and lemma III.4. The fact that LLS estimation in the time domain maps to LLS estimation in the DFT transform domain allows the coding theorems in theorem III.3 and theorem II.5 to apply, hence theorem III.5 follows immediately. □

IV. CONCLUSION

Our results points to the efficient implementation of distributed processing in channel coding, via two dimensional (2D) Kalman filter, when we assume that the channel is additive white Gaussian noise, single input multi output, and has memory (intersymbol interference). We also assume the transmitter and receiver have complete knowledge of the channel. By efficient we mean that the capacity of the channel with distributed processing is the same as that of the channel with centralized processing, and moreover the processing, that is, either maximum likelihood (ML) or Bayes least square (BLS) or linear least square (LLS) estimation is done successively in time-space and in a distributed manner.

When the capacity achieving input process (a wide sense stationary stochastic process) has infinite memory, a truncation strategy would be required. The strategy is geared towards finding the most sensibly finite amount of memory to assign in the state vector of the state-space estimation model for the 2D Kalman filtering such that the error covariance gap between the centralized LLSE (can be ML or BLS) estimate of the state vector and the successive estimate of state vector by means of 2D Kalman filtering is reasonably small. This problem is not elaborated in this paper and is a subject for further study.

Still geared towards the question of whether distributed processing yields achievable capacity equal to that of centralized processing, an interesting channel model to study is a channel that is not perfectly known at the transmitter, namely, the receiver estimates the channel with some error and makes the estimate known to the transmitter.

REFERENCES

[1] B.D.O. Anderson and J.B. Moore: *Optimal Filtering*, Prentice Hall, Englewood Cliffs, NJ, 1979

[2] T. Berger, Z. Zhang, and H. Viswanathan: *The CEO Problem*, IEEE Trans. Inf. Theory Vol. 42, pp. 887-902, (May '96)

[3] T.M. Cover and J.A. Thomas: *Elements of Information Theory*, John Wiley & Sons, Inc., 1991

[4] S.C. Draper: *Successive Structuring of Source Coding Algorithms for Data Fusion, Buffering, and Distribution in Networks*, PhD Thesis, MIT, 2002

[5] R.G. Gallager: *Information Theory and Reliable Communication*, John Wiley & Sons, Inc., 1968

[6] W. Hirt and J.L. Massey: *Capacity of the Discrete-Time Gaussian Channel with Intersymbol Interference*, IEEE Trans. Inf. Theory Vol. 34, No. 3 ('88)

[7] M. Médard: *Processing of Wireless Signals to Preserve Wireline Network Resources*, Wireless Communications and Networking Conference (WCNC) Vol. 1, pp.61 -65, ('99)
[8] Y. Oohama: *The Rate-Distortion Function for the Quadratic Gaussian CEO Problem*, IEEE Trans. Inf. Theory Vol. 44, pp. 1057-1070, (May '98)
[9] H. Viswanathan and T. Berger: *The Quadratic Gaussian CEO Problem*, IEEE Trans. Inf. Theory Vol. 43, pp. 1549-1559, (September '97)
[10] A.S. Willsky, G.W. Wornell, J.H. Shapiro: *Course Notes on Stochastic Processes, Detection, and Estimation*, Mass. Inst. Technol., Cambridge, MA, Sept. 1998