THE STATE SPACE FRAMEWORK FOR BLIND DYNAMIC SIGNAL EXTRACTION AND RECOVERY

F. M. Salam

Department of Electrical Engineering
Michigan State University
East Lansing, Michigan 48824-1226, USA

G. Erten

IC Tech, Inc.
2157 University Park Dr.
Okemos, Michigan 48864, USA

ABSTRACT

The paper describes a framework in the form of an optimization of a performance index subject to the constraints of a dynamic network, represented in the state space. The performance index is a measure of statistical dependence among the outputs of the network, namely, the relative entropy also known as the Kullback-Leibler divergence. The network is represented as (either discrete or continuous time) state space dynamics. Update laws are derived in the general cases. Moreover, in the discrete-time case, they are shown to specialize to the FIR and IIR network representations.

1. INTRODUCTION

Blind signal processing is a critical area with a wide applications. Formulation with real-world and general environments are currently under investigations by numerous researchers (see [1-6] and the references therein for a sample).

The state space approach formulation of the signal separation, extraction, and recovery was formulated in [1] for linear time-invariant systems. The state notion summarizes delays and filtering of driving signals or source.

Recent works have used the Kullback-Leibler divergence as a measure for statistical independence [2]. The state space formulation for linear environments have been introduced in [1,5] and the references therein. The work here emphasizes the general formulation, specifically for discrete-time nonlinear state space systems, and shows how to specialize the linear time-invariant case to the FIR and IIR filters. The general continuous-time case is analogous and has been introduced in [6]. Further details are delegated to [7].

2. ADAPTIVE NETWORKS FOR BLIND SIGNAL RECOVERY

The "mixing" environment may be represented by an unknown process with inputs being the independent sources and outputs being the measurements. In this extreme case, no structure is assumed about the model of the environment.

The environment may be modeled as a dynamic system with fixed but unknown parameters. The processing network must be constructed with the capability to compute the "inverse" (or the "closest" to an inverse) of the environment model.

It is possible that an augmented network be constructed so that the inverse of the environment is merely a subsystem of the network with learning. And hence, even if the inverse of the environment is unstable (say because of non-minimum phase zeros), the overall augmented network with the update laws represent a nonlinear adaptive dynamic system which may converge to the parameters as a stable equilibrium point. Thus identification of the process is achieved.

The Performance Measure/Functional

The mutual information of a random vector **y** is a measure of dependence among its components and is defined as ([2, 4]):

The continuous case:

$$L(y) = \int_{y \in Y} p_y(y) \quad \ln\left\{\frac{p_y(y)}{\prod_{j=1}^{j=r} p_{y_j}(y_j)}\right\} dy$$

The discrete case:

$$L(y) = \sum_{y \in Y} p_y(y) \quad \ln\left\{\frac{p_y(y)}{\prod_{j=1}^{j=r} p_{y_j}(y_j)}\right\}$$

An approximation of the discrete case:

$$L(y) \cong \sum_{k=k_0}^{k_1} p_y(y(k)) \quad \ln\left\{\frac{p_y(y(k))}{\prod_{j=1}^{j=r} p_{y_j}(y_j(k))}\right\}$$

where $p_y(y)$ is the probability density function (pdf) of the random vector **y,** whereas $p_{y_j}(y_j)$ is the probabilty density of the j-th component of the output vector **y.** The functional L (**y**) is always non-negative and is zero if and only

if the components of the random vector **y** are statistically independent. This important measure defines the *degree of dependence* among the components of the signal vector. Therefore, it represents an appropriate functional for characterizing (the degree of) statistical independence. *L(**y**)* can be expressed in terms of the entropy

$$L(\mathbf{y}) = -H(\mathbf{y}) + \sum_i H(y_i)$$

where H (**y**) := - E[ln $f_\mathbf{y}$], is the **entropy** of **y**, and E[.] denotes the expected value.

The General Nonlinear Dynamic Case:

The Environment Model:

Let the environment be modeled as the following nonlinear discrete-time dynamic (forward) processing model:

$$X_p(k+1) = f_p(X_p(k), s(k), w_1^*)$$
$$m(k) = g_p(X_p(k), s(k), w_2^*)$$

where s(k) is an n-dimensional vector of original sources, m(k) is the m-dimensional vector of measurements, X_p(k) is the N_p-dimensional state vector. The vector (or matrix) w_1^* represents constant/parameter of the dynamic equation, and w_2^* represents constant/parameter of the "output" equation. The functions $f_p(.)$ and $g_p(.)$ are differentiable. It is also assumed that existence and uniqueness of solutions of the differential equation are satisfied for each set of initial conditions $X_p(t_0)$ and a given waveform vector s(k).

The Processing Networks:

The (processing) network may be represented by a dynamic (forward) network or a dynamic feedback network [4-7].

The Feedforward Network is

$$X(k+1) = f(X(k), m(k), w1)$$
$$y(k) = g(X(k), m(k), w2)$$

where m(k) is the m-dimensional measurement, y(k) is the r-dimensional output vector, $X(k)$ is the N-dimensional state vector. (Note that N and N_p may be different.) The vector (or matrix) w_1 represents the parameter of the dynamic equation, and w_2 represents the parameter of the "output" equation. The functions $f(.)$ and $g(.)$ are differentiable. It is also assumed that existence and uniqueness of solutions of the differential equation are satisfied for each set of initial conditions $X(t_0)$ and a given measurement waveform vector m(k).

Update Law for The Feedforward architecture:

The Discrete-time dynamic network: general nonlinear case

The update law is now developed for dynamic environments to recover the original signals following the procedures in [4-7].

The Feedforward Case:

The network is a feedforward dynamical system. In this case, one defines the performance index

$$J_0(w_1, w_2) = \sum_{k=k_0}^{k_1-1} L^k(y_k)$$

subject to the discrete-time nonlinear dynamic network

$$X_{k+1} = f^k(X_k, m_k, w_1), \qquad X_{k_0}$$
$$y_k = g^k(X_k, m_k, w_2)$$

Then, the augmented cost function to be optimized becomes

$$J_0'(w_1, w_2) = \sum_{k=k_0}^{k_1-1} L^k(y_k) + \boldsymbol{1}_{k+1}^T(f^k(X_k, m_k, w_1) - X_{k+1})$$

The Hamiltonian is then defined as

$$H^k = L^k(y(k)) + \boldsymbol{1}_{k+1}^T f^k(X, m, w_1)$$

Consequently, the sufficient conditions for optimality are:

$$X_{k+1} = \frac{\partial H^k}{\partial \boldsymbol{1}_{k+1}} = f^k(X_k, m_k, w_1)$$

$$\boldsymbol{1}_k = \frac{\partial H^k}{\partial X_k} = (f_{X_k}^k)^T \boldsymbol{1}_{k+1} + \frac{\partial L^k}{\partial X_k}$$

$$\Delta w_1 = -\boldsymbol{h}\frac{\partial H^k}{\partial w_1} = -\boldsymbol{h}(f_{w_1}^k)^T \boldsymbol{1}_{k+1}$$

$$\Delta w_2 = -\boldsymbol{h}\frac{\partial H^k}{\partial w_2} = -\boldsymbol{h}\frac{\partial L^k}{\partial w_2}$$

The general Discrete Linear Dynamic Case:

Environment model:

$$\overline{X}_p(k+1) = \overline{A}\ \overline{X}_p(k) + \overline{B}\ s(k)$$
$$m(k) = \overline{C}\ \overline{X}_p(k) + \overline{D}\ s(k)$$

Feedforward Network

$$X(k+1) = A\ X(k) + B\ m(k)$$
$$y(k) = C\ X(k) + D\ m(k)$$

The first question is the following: Does there exist parameter matrices of the processing network which would recover the original signals? The answer is yes, the explicit solutions of the parameters are given in [4-7].

Canonical Representations and the State Space Specialization to Discrete-time IIR and FIR Filters

The general discrete-time linear dynamics of the network are given as:

$$X(k+1) = A\,X(k) + B\,m(k)$$
$$y(k) = C\,X(k) + D\,m(k)$$

where m(k) is the m-dimensional vector of measurements, y(k) is the n-dimensional vector of (processed) outputs, and X(k) is the (mL) dimensional states (representing filtered versions of the measurements in this case). One may view the state vector as composed of he L m-dimensional state vectors $X_1, X_2, \ldots, X_L$. That is,

$$X_k = X(k) = \begin{bmatrix} X_1(k) \\ X_2(k) \\ \ldots \\ X_L(k) \end{bmatrix}$$

<u>Special Case:</u>

Consider the case where the matrices and A and B are in the "controllable canonical form." We represent the A and B block matrices as,

$$A = \begin{bmatrix} A_{11} & A_{12} & \ldots & A_{1L} \\ I & 0 & \ldots & 0 \\ \ldots & I & \ldots & 0 \\ 0 & 0 & I & 0 \end{bmatrix}, \quad \text{and} \quad B = \begin{bmatrix} I \\ 0 \\ . \\ 0 \end{bmatrix}$$

where each block sub-matrix A_{1j} may be simplified to a diagonal matrix, and each I is a block identity matrix with appropriate dimensions.

Then,

$$X_1(k+1) = \sum_{j=1}^{L} A_{1j}\,X_j(k) + m(k)$$
$$X_2(k+1) = X_1(k)$$
$$\ldots$$
$$X_L(k+1) = X_{L-1}(k)$$

$$y(k) = \sum_{j=1}^{L} C_j X_j(k) + D\,m(k)$$

This model represents an IIR filtering structure of the measurement vector m(k). In the event that the block matrices A_{1j} are zero, then the model is reduced to the special case of an FIR filter.

$$X_1(k+1) = m(k)$$
$$X_2(k+1) = X_1(k)$$
$$\ldots$$
$$X_L(k+1) = X_{L-1}(k)$$

$$y(k) = \sum_{j=1}^{L} C_j X_j(k) + D\,m(k)$$

The equations may be re-written as

$$X_1(k) = m(k-1)$$
$$X_2(k) = X_1(k-1) = m(k-2)$$
$$\ldots$$
$$X_L(k) = X_{L-1}(k-1) = m(k-L)$$

$$y(k) = \sum_{j=1}^{L} C_j X_j(k) + D\,m(k)$$

<u>The Update law for the linear dynamic case:</u>

The Discrete-time dynamic network: The linear time-invariant case

$$X_{k+1} = \frac{\partial H^k}{\partial \mathbf{l}_{k+1}} = f^k(X, m, w_1) = AX_k + Bm_k$$

$$\mathbf{l}_k = \frac{\partial H^k}{\partial X_k} = (f_{X_k}^k)^T \mathbf{l}_{k+1} + \frac{\partial L^k}{\partial X_k} = A_k^T \mathbf{l}_k + C_k^T \frac{\partial L^k}{\partial y_k}$$

$$\Delta A = -\mathbf{h}\frac{\partial H^k}{\partial A} = -\mathbf{h}(f_A^k)^T \mathbf{l}_{k+1} = -\mathbf{l}_{k+1} X_k^T$$

$$\Delta B = -\mathbf{h}\frac{\partial H^k}{\partial B} = -\mathbf{h}(f_B^k)^T \mathbf{l}_{k+1} = -\mathbf{l}_{k+1} m_k^T$$

$$\Delta D = -\mathbf{h}\frac{\partial H^k}{\partial D} = -\mathbf{h}\frac{\partial L^k}{\partial D} = \mathbf{h}([D]^{-T} - f_\mathbf{a}(y)\,m^T)$$

$$\Delta C = -\mathbf{h}\frac{\partial H^k}{\partial C} = -\mathbf{h}\frac{\partial L^k}{\partial C} = \mathbf{h}(-f_\mathbf{a}(y)\,X^T)$$

<u>Special Canonical Representation cases:</u>

The matrices A and B are best represented in the "controllable canonical forms" or the form I format. That is, B is constant and A has only the first block rows as parameters in the IIR network case. In that event, No update equation for the matrix B are used. While for the matrix A only the first block rows are updated. Thus the update law for the matrix A is limited to

$$\Delta A_{1j} = -\mathbf{h}\frac{\partial H^k}{\partial A_{1j}} = -\mathbf{h}(f_{A_{1j}}^k)^T \mathbf{l}_{k+1}$$

$$= -\mathbf{h}\mathbf{l}_1(k+1) X_j^T(k)$$

Noting the form of the matrix A, the co-state equations can be expanded as

$$\boldsymbol{l}_1(k) = \boldsymbol{l}_2(k+1) + C_1^T \frac{\partial L^k}{\partial y_k}(k)$$

$$\boldsymbol{l}_2(k) = \boldsymbol{l}_3(k+1) + C_2^T \frac{\partial L^k}{\partial y_k}(k)$$

$$\vdots$$

$$\boldsymbol{l}_L(k) = C_L^T \frac{\partial L^k}{\partial y_k}(k)$$

which produce the solutions at time k and k+1

$$\boldsymbol{l}_1(k) = \sum_{l=1}^{L} C_l^T \frac{\partial L^k}{\partial y_k}(k+l-1)$$

$$\boldsymbol{l}_1(k+1) = \sum_{l=1}^{L} C_l^T \frac{\partial L^k}{\partial y_k}(k+l)$$

Therefore, the update law for the block sub-matrices in A are:

$$\Delta A_{1j} = -\boldsymbol{h}\frac{\partial H^k}{\partial A_{1j}} = -\boldsymbol{h}\boldsymbol{l}_1(k+1)X_j^T(k)$$

$$= -\boldsymbol{h}\sum_{l=1}^{L} C_l^T \frac{\partial L^k}{\partial y_k}(k+l)\,X_j^T$$

The [D] represents the transpose of the pseudo-inverse of the D matrix. The update laws for the matrices D and C can be elaborated upon as follows:

$$\Delta D = \boldsymbol{h}([D]^{-T} - f_{\boldsymbol{a}}(y)\,m^T) = \boldsymbol{h}(I - f_{\boldsymbol{a}}(y)\,(Dm)^T)[D]^{-T}$$

where I is a matrix composed of the rxr identity matrix augmented by additional zero row (if n> r) or additional zero columns (if n < r). In light of considering the "natural gradient," an alternate update law in this case is

$$\Delta D = \boldsymbol{h}([D]^{-T} - f_{\boldsymbol{a}}(y)\,m^T)D^T D = \boldsymbol{h}(I - f_{\boldsymbol{a}}(y)\,(Dm)^T)D$$

For the C matrix, the update equations can be written for each block matrix as follows:

$$\Delta C_j = -\boldsymbol{h}\frac{\partial H^k}{\partial C_j} = -\boldsymbol{h}\frac{\partial L^k}{\partial C_j} = \boldsymbol{h}(-f_{\boldsymbol{a}}(y)\,X_j^T)$$

SUMMARY

The formulation of the blind signal processing in the context of optimization subject to dynamic constraints is developed and shown of the general nonlinear and linear discrete-time cases. In building on our previous work focusing on continuous-time dynamic modeling, this complete the derivations for both continuous-time and discrete time modeling of the "mixing" environment. Application of this theory in simulation and in real-world environments has shown considerable improvements in performance [7]. Audio results will be presents at the conference in the pre-conference tutorial. Further details are deferred to [7].

REFERENCES

[1] F. M. Salam. An Adaptive Network for Blind Separation of Independent Signals. Proceedings of the 1993 IEEE International Symposium on Circuits and Systems (ISCAS) Vol.I, pp. 431 - 434.

[2] S. Amari, A. Cichocki, and H. Yang. A new learning algorithm for blind signal separation, MIT Press, 1996.

[3] L.Tong, Y. Inouye, and R. Lio. Wave-preserving blind estimation of multiple independent sources. IEEE Trans. On Signal Processing, 41 (7): 2461-2470, July 1993.

[4] F. Salam, A. Gharbi, and G. Erten, "Formulation and adaptation For Signal Separation in Dynamic Environments," The IEEE 40th Midwest Symposium on Circuits and Systems, Sacramento, CA, 4-6 August 1997.

[5] G. Erten and F. Salam, "Real Time Separation of Audio Signals Using Digital Signal Processors, " MWSCAS 97 Sacramento, California, USA, August 1997.

[6] F. Salam and G. Erten, "Blind Signal Separation and Recovery in Dynamic Environments," 1997 IEEE Workshop on Nonlinear Signal and Image Processing, Mackinac Island, September 8-10, 1997.

[7] F. Salam and G. Erten, Optimization with constraints formulation of the blind Signal Recovery, M-report 1998-12.