KALMAN FILTERING FOR SELF-SIMILAR PROCESSES

Meltem Izzetoğlu, Birsen Yazıcı, Banu Onaral, Nihat Bilgütay

Drexel University,
Electrical and Computer Eng. Dept.,
Philadelphia, PA, 19104

ABSTRACT

In our earlier work, we introduced a class of stochastic processes obeying a structure of the form, $E[X(t)X(t\lambda)] = R(\lambda)$, $t, \lambda > 0$, and outlined a mathematical framework for the modeling and analysis for these processes. We referred to this class of processes as scale stationary processes. We demonstrated that scale stationarity framework leads to engineering oriented mathematical tools and concepts, such as autocorrelation and spectral density function and finite parameter ARMA models for modeling and analysis of statistically self-similar signals. In this work, we will introduce a state space representation for self-similar signals and systems based on scale stationary ARMA models. Such a representation provides a complete description of the inner and outer dynamics of a self-similar system or signal that can not be obtained from transfer function representation. We will introduce Kalman filtering techniques and Ricatti Equations for smoothing and prediction of self-similar processes.

1. INTRODUCTION

$1/f$ processes occur in a broad range of engineering and science applications including network traffic, noises in electronic devices, biomedical systems, burst error in communication channels to mention a few [1]-[9].

The major characteristics of these processes are their long term correlation structure, and their statistical self-similarity. These characterizations are apparent in the empirical $1/f^\gamma$ power spectrum. Typically, the parameter γ controls both the degree of long term correlations and the statistical self-similarity. Mathematical tools and concepts for such processes were first formulated and advocated in practice by Mandelbrot [2] within the context of "fractals". He proposed the well-known fractional Brownian motion (fBm) model to capture the long term correlation and statistical self-similarity of the $1/f$ processes. Given the elaborate fBm model, and the aura of "fractal science", a flurry of activity evolved around the modeling and analysis of $1/f$ processes in engineering literature [3]-[6]. However, these efforts never hold a strong ground in engineering applications, mainly due to the mathematical intractability of the fBm model, and the lack of foundational principles. In [1], Yazıcı et al. proposed a class of second order processes obeying a structure of the form $E[X(t)X(t\lambda)] = R(\lambda)$, $t, \lambda > 0$ to model and analyze $1/f$ processes. These models, referred to as scale stationary, enjoy theoretical properties parallel to the ordinary wide sense stationary processes. Most importantly, their foundation is based on the extensions of the concept of stationarity on which powerful time series analysis tools are derived. Scale stationary processes come with the spectral analysis tools, and ARMA models just like the ordinary stationary processes. They are also directly linked to the linear scale invariant systems. Let us not forget to mention that, fBm model is simply a trended scale stationary model with stationary increments. It may be academically dissapointing! but true that the issue of "sta-

tistical self-similarity" can be managed to a large degree by the simple framework of "scale stationarity".

In [1], authors introduced scale stationary ARMA models based on Euler-Cauchy system and showed that any scale stationary process can be captured by a finite parameter scale stationary autoregressive model. In this study, we extend the ARMA modeling to multiple input and multiple output (MIMO) systems and propose a state space representation for the self-similar processes. At first glance, both the state space representation and the Kalman filter may appear simply as time-varying models. However, with the proper definition of the derivative operation on the multiplicative group and the self-similarity, both the state space model and the Kalman filter are captured with constant matrix vector representation. This new definition of the derivative operation guides the implementation of the Kalman filter for self-similar processes, both in recursive update and the Ricatti equation, leading to superior performance than the ordinary time varying implementation.

The proposed state space representation and the Kalman filtering can be used in estimation, and prediction tasks involving $1/f$ phenomena. Applications include inverse filtering for communication channels and blurred images in which the blur or the channel is time varying and the underlying data and noise have $1/f$ characteristics. Another obvious application of the tool is in communication network traffic prediction which has potential implications in network management and quality service provisioning.

The organization of the paper is as follows: Section 2 covers the basic background on scale stationary processes and scale stationary ARMA modeling. Section 3 presents the state space representation and the derivative operator for functions defined on the multiplicative group. Section 4 introduces the Kalman filter for self-similar processes. Section 5 discusses the implementation of he Kalman filter and the Ricatti Equation and presents some simulation results. Section 6 discusses the applications of the proposed Kalman filter in various engineering problems. Finally, Section 7 concludes the discussion.

2. BACKGROUND ON SELF-SIMILAR PROCESSES

Before giving the derivation of our state space model and Kalman filtering, we like to summarize related background information on self-similar processes as introduced in detail in [1]. A linear system satisfying

$$S\{x(t\lambda)\} = \lambda^{-H}y(t\lambda) \qquad (1)$$

is called a Linear Self-Similar (LSS) system with self-similarity parameter H. As it can be seen from this definition, analogous to LTI systems which are invariant to time shifts, LSS systems are invariant to scale changes within a constant parameter.

The output of the LSS system to any input is found by a scale convolution operation defined as:

$$y(t) = \tilde{h}(t) * u(t) = t^H \int_0^\infty \tilde{h}(\frac{t}{\lambda})u(\lambda)dln\lambda, \qquad t0 \qquad (2)$$

where $t^H \tilde{h}(t)$ is the response of the system to the *unit driving force*, $\tilde{\delta}(t)$ [1] defined as: i) $\tilde{\delta}(t) = 0, t \neq 1$ $t > 0$, ii) $\int_0^\infty \tilde{\delta}(t/\lambda)dln\lambda = 1, t > 0$, iii) $x(t) = \int_0^\infty x(\lambda)\tilde{\delta}(\frac{t}{\lambda})dln\lambda$.

A linear dynamical model for LSS systems is represented as time varying Euler-Cauchy type differential equations:

$$a_N t^N \frac{d^N}{dt_N}y(t) + \cdots + a_1 t\frac{d}{dt}y(t) + a_0 y(t) =$$

$$b_M t^{M+H} \frac{d^M}{dt_M}u(t) + \cdots + b_1 t^{1+H} \frac{d}{dt}u(t) + b_0 t^H u(t) \quad (3)$$

This type of system satisfies the self-similarity definition as in (1). The difference of the Euler-Cauchy system actually comes from the fact that the dynamics of the system is captured in scale derivatives defined on the multiplicative group [10] as:

$$t\frac{d}{dt}y(t) = \lim_{\Delta \to 1} \frac{y(t\Delta) - y(t)}{ln\Delta} \qquad (4)$$

Since the model is invariant to scale changes, the memory of the system is stored in infinitesimal time scalings, similar to the Euler dynamical model for LTI systems where the memory of the system is stored in infinitesimal time shifts since these systems are time invariant.

In a probabilistic setting Euler-Cauchy system generates self-similar processes with $1/f$ spectrum [1]. Using the input-output relationship of the LSS system (2), the power spectrum of the Euler-Cauchy system in Fourier domain driven by a white noise having autocorrelation $R_u(t_1, t_2) = \sigma^2 \tilde{\delta}(t_2/t_1)$ is shown to have power-law or a $1/f$ spectrum [1]. Therefore, any $1/f$ process can be approximated with a finite order Euler-Cauchy system which makes signal processing techniques for estimation and prediction of such processes possible. It is because of these facts that we use Euler-Cauchy systems in the derivation of the state space representation and Kalman filtering algorithm for LSS systems.

3. STATE SPACE REPRESENTATION OF SELF-SIMILAR PROCESSES

Beginning from the Euler-Cauchy system in (3), the general state space representation with states having different self-similarity parameters can be obtained as:

$$t\frac{d}{dt}\mathbf{x}(t) = t^H(\mathbf{A} + \mathbf{H})t^{-H}\mathbf{x}(t) + t^H\mathbf{B}u(t) \qquad (5)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}u(t) \qquad (6)$$

where $\mathbf{x}(t) = [x_1(t)\ x_2(t)\ ...\ x_N(t)]^T$ ($[.]^T$ is the transpose operation) is the $Nx1$ state vector, $\mathbf{u}(t)$ is the $Rx1$ input vector, $\mathbf{y}(t)$ is the $Mx1$ output vector, $\mathbf{A}$ is a NxN matrix, $\mathbf{B}$ is a NxR matrix, $\mathbf{C}$ is a MxN matrix, $\mathbf{D}$ is a MxR matrix and $\mathbf{H}$ is a NxN diagonal matrix having values $H_1, H_2, ..., H_N$ in its diagonal entries.

In this representation, the self-similarity parameters, H_i for $i = 1, ..., N$ of the states can be equivalent, then the external system representation reduces to the Euler-Cauchy system in (3). However, for the states to have same self-similarities is not very realistic and it is a specific case of the general form. Therefore, we use the most general state space representation with the states having different H_is throughout the paper.

It can be argued that the state space representation for LSS systems can be expressed as first order time varying ordinary differential equations $\frac{d}{dt}\mathbf{x}(t) = \frac{t^H(\mathbf{A}+\mathbf{H})t^{-H}t}{t}\mathbf{x}(t) + \frac{t^H\mathbf{B}}{t}\mathbf{u}(t)$ and time

varying state space techniques can be used in their analysis. In this type of representation, the memory of the states are captured in infinitesimal time shifts as in the LTI systems. However, here for the LSS systems, expressing the inner dynamics of the whole system with first order self-similar Euler-Cauchy systems as states is more appropriate to the nature of the dynamics of the system. This is because of the fact that the states are also self-similar in nature therefore, their energy should be stored in infinitesimal time scalings as in (4).

In order to analyze the self-similar dynamics of the states more closely, let us consider the general kth state, $x_k(t)$ whose dynamical equation is:

$$t\frac{d}{dt}x_k(t) = (a_{k,k} + H_k)x_k(t) + t^{H_k}\sum_{\substack{l=1 \\ l \neq k}}^N a_{k,l}t^{-H_l}x_l(t) +$$

$$t^{H_k}\mathbf{B}u(t) \qquad (7)$$

As can be seen from this equation, the dynamics of a state is affected by the state itself, other states and the input depending on the $\mathbf{A}$ and $\mathbf{B}$ matrices. The dependency on the state itself can be seen as an intrinsic self-similarity since the self-similarity parameter appears as a constant gain factor. If there is coupling to the other states, these states can be treated as inputs where the self-similarity of the state is provided with the fractional or self-similar leakage term t^{H_k}. Note here that the self-similarity of the coupled states $x_l(t)$ for $l = 1, ..., N$ and $l \neq k$ with parameters H_l does not have an effect on the self-similarity of the state $x_k(t)$ which guarantees a self-similar first order system for each state $x_k(t)$ with only one self-similarity parameter, H_k.

The solution of the states can be found using the state transition matrix $\Phi(t, \tau)$ as:

$$\mathbf{x}(t) = \Phi(t, t_1)\mathbf{x}(t_1) + \int_{t_1}^t \Phi(t, \tau)\tau^H\mathbf{B}u(\tau)dln\tau \qquad (8)$$

where $\Phi(t, \tau)$ can be obtained using the fundamental matrix $\Phi(t) = t^H t^\mathbf{A}$ which is a solution of the homogeneous state equation in (5) as:

$$\Phi(t, \tau) = \Phi(t)\Phi^{-1}(\tau) = t^H(\frac{t}{\tau})^\mathbf{A}(\tau)^{-H} \qquad (9)$$

Note here that the state transition matrix is also a solution of the homogeneous state equation and it satisfies the same properties as it LTI counterpart [12]. The unit driving force response is found as $\tilde{h}(t, \tau) = t^H(\frac{t}{\tau})^\mathbf{A}\mathbf{B}$ using (8). Then the solution for the states and the outputs are:

$$\mathbf{x}(t) = t^H t^\mathbf{A}\mathbf{x}(t_1) + t^H \int_{t_1}^t (\frac{t}{\tau})^\mathbf{A}\mathbf{B}u(\tau)dln\tau$$

$$\mathbf{y}(t) = \mathbf{C}t^H t^\mathbf{A}\mathbf{x}(t_1) + \mathbf{C}t^H \int_{t_1}^t (\frac{t}{\tau})^\mathbf{A}\mathbf{B}u(\tau)dln\tau \quad (10)$$

Although each state $x_k(t)$ for $k = 1, ..., N$ is self-similar with self-similarity parameter H_k, depending on the matrix $\mathbf{C}$, the outputs $y_j(t)$ for $j = 1, ..., M$ can be expressed with either one self-similar state or a linear combination of self-similar states with different self-similarity parameters.

4. KALMAN FILTERING

In this section we will investigate the problem of estimating the state variables of a self-similar process by using noisy measurements of the linear combination of the states. Consider the LSS system in state space:

$$t\frac{d}{dt}\mathbf{x}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{w}(t) \qquad (11)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{v}(t) \qquad (12)$$

where $A(t) = t^{H}(A + H)t^{-H}$ and $B(t) = t^{H}B$. Equation (11) is the system model where $w(t)$ is the system noise and equation (12) is the measurement model where $v(t)$ is the measurement noise. Both $w(t)$ and $v(t)$ are zero mean white Gaussian noise with covariances:

$$E\{w(t)w^{T}(\tau)\} = Q(t)\tilde{\delta}(t/\tau)$$

$$E\{v(t)v^{T}(\tau)\} = R(t)\tilde{\delta}(t/\tau) \qquad (13)$$

They are also uncorrelated with each other and the states.

Assuming that $A(t), B(t), C$ and H are completely known, the state estimate $\hat{x}(t)$ is obtained by feeding a correction term back to the estimated system depending on the difference between the actual measurement and the estimated measurement as:

$$t\dot{\hat{x}}(t) = t^{H}(A + H)t^{-H}\hat{x}(t) + K(t)(y(t) - C\hat{x}(t)) \qquad (14)$$

where $K(t)$ is the Kalman gain matrix that has to be estimated optimally.

Then the error states between the actual and estimated states $\tilde{x}(t) = \hat{x}(t) - x(t)$ satisfy:

$$t\dot{\tilde{x}}(t) = (A(t) - K(t)C)\tilde{x}(t) + K(t)v(t) - B(t)w(t) \qquad (15)$$

Using the solution of the error state in terms of its state transition matrix $\Phi_{\tilde{x}}(t,\tau)$, the covariance of the error state $P(t)$ is:

$$P(t) = E\{\tilde{x}(t)\tilde{x}^{T}(\tau)\} = \Phi_{\tilde{x}}(t,t_{1})P(t_{1})\Phi_{\tilde{x}}^{T}(t,t_{1}) +$$
$$\int_{t_{1}}^{t} \Phi_{\tilde{x}}(t,\tau)K(\tau)R(\tau)K^{T}(\tau)\Phi_{\tilde{x}}^{T}(t,\tau)dln\tau +$$
$$\int_{t_{1}}^{t} \Phi_{\tilde{x}}(t,\tau)B(\tau)Q(\tau)B^{T}(\tau)\Phi_{\tilde{x}}^{T}(t,\tau)dln\tau \qquad (16)$$

where $P(t_{1})$ is the initial error covariance matrix at initial time t_{1}.

For the estimated state to be optimal, the error should be minimized in time via $K(t)$. In order to find the optimum Kalman filter gain, the cost function $J(t)$ related to the error state as:

$$J(t) = E\{\tilde{x}^{T}(t)\tilde{x}(t)\} = Trace\{P(t)\} \qquad (17)$$

should be minimized in the MMSE sense.

After some manipulations as explained in [11] the Kalman gain matrix that minimizes the cost function is found as:

$$K(t) = PC^{T}R^{-1}(t) \qquad (18)$$

Then using this $K(t)$ in (16), the change in the error covariance or the Riccati equation can be obtained as:

$$t\dot{P}(t) = A(t)P(t) + P(t)A^{T}(t) + B(t)Q(t)B^{T}(t) - K(t)R(t)K^{T}(t) \qquad (19)$$

Here, the Kalman filtering algorithm has the same structure as its LTI counterpart. The major difference of our algorithm from the LTI case lies in the state update (14) and error covariance propagation equations (19). Here, the memory is captured in infinitesimal time scalings instead of time shiftings as opposed to the LTI case. Therefore, the self-similar nature of the state estimate and error covariance is satisfied.

5. IMPLEMENTATION AND SIMULATION RESULTS

We simulate a first order LSS system with parameters $H = -0.2$, $A = -0.1, B = 0.1, C = 1, Q = 1$ to test the performance of the proposed Kalman filter.

We generate the $1/f$ data, $x(t)$ via a covariance method that uses Karhunen-Loeve (KL) transform. The autocovariance of $x(t)$ for the first order system given above

$$C_{xx}(t_{1}, t_{2}) = \beta(t_{1}t_{2})^{(A+H)}(max(t_{1},t_{2})^{(-2A)} - 1); \qquad (20)$$

where $\beta = B^{2}Q/(-2A)$. Then using this covariance matrix and KL transform we generate the $1/f$ data, $x(t)$ for $1 < t < 20$. In the Kalman filtering algorithm, the estimated state in continuous time is approximated using the scale derivative definition (4) in geometric time intervals:

$$\hat{x}(\Delta t) = \hat{x}(t) + ln\Delta(A\hat{x}(t) + K(t)(y(t) - C\hat{x}(t))) \qquad (21)$$

and the Riccati equation solution:

$$A(t)P(t) + P(t)A^{T}(t) + B(t)Q(t)B^{T}(t) - K(t)R(t)K^{T}(t) = 0 \qquad (22)$$

is obtained using the Schur algorithm as given in [13]. The continuous time approximation becomes more accurate when the scale step Δ is selected as close to 1 as possible. Here, in our application we select it as $\Delta = 1.01$.

We test the performance of the Kalman filter for two different SNRs of 20, and 10dBs using 100 Monte Carlo Runs (MCR). The SNR of the signal is calculated as:

$$SNR = var(x)/var(v) \qquad (23)$$

A sample data $x(t)$ (solid line), $y(t)$ (dash-dot line) and the estimated data $\hat{x}(t)$ (dashed line) for $SNR = 20$ and 10dBs out of 100 MCR are given in Figure 1 and 2, respectively.

Then the estimation SNR' for each estimated signal $\hat{x}(t)$ is calculated as:

$$SNR' = var(x)/var(x - \hat{x}) \qquad (24)$$

For the input $SNR = 20$ and 10dB, the range of estimation SNR's for 100 MCR in each case are found as 7.73dB$< SNR' <$ 8.15dB and 3.02dB$< SNR' <$ 3.69dB where the mean values of them are 7.92dB and 3.36db, respectively.

Let us mention that the proposed Kalman filter also suffers from the same problems as the usual Kalman filter, such as the building up of the "random walk" type error as the prediction time increases. This problem can be overcome with the usage of a backward smoother if the offline processing is possible.

6. APPLICATION AREAS

In this section, we will explain two potential application areas of the proposed Kalman filtering procedure 1) packet arrival estimation in self-similar network traffic and 2) time varying fading channel estimation during self-similar signal transmission in wireless communication applications.

Network traffic studies show that the aggregate of the packet arrival shows the same statistics of long range correlations which decays hyperbolically, the variance of the sample mean decays slowly and their power spectrum obey power law near the origin over different time scales. This observation is apparently valid for Ethernet traffic, ISDN packet networks, signaling (CCSN/SS7) networks for public telephone networks, [7, 8]. If $x_{1}, x_{2}, x_{3}, ...$ denote the

number of arrivals in the first, second,... interval, the aggregate of these arrivals in consecutive, non-overlapping block of m intervals are calculated as follows: Let x_1^m denote the mean arrival rate of the first m intervals $(x_1 + x_2 + ... + x_m)/m$, x_k^m denote $(x_{m(k-1)+1} + ... + x_{km})/m$ and so on. Actually these aggregate processes give the whole arrival process in different time scales and since the aggregate arrival process shows the same long range statistics with slowly decaying variances and self similar characteristic, it is a self-similar process as opposed to the early assumptions of Poisson distribution. Therefore, analysis techniques for traffic density must consider this self-similar nature. Especially, since the buffering requirements for self-similar processes are larger than that are estimated with Poisson processes, the techniques should be selected carefully for the estimation of buffer size. Using the proposed Kalman filtering technique, the self-similar data traffic can be predicted recursively.

Another application area for the proposed Kalman filter is in communications. In present wireless communication applications such as radar, sonar, acoustics, etc., the transmission channel is usually modeled as a multipath fading channel having slowly time-varying characteristics. At the receiver end the transmitted signal through a multipath fading channel is further corrupted by noise. It is an important and a difficult task to deconvolve the original signal from this received data, especially when the transmitted signal and the corruption noise are nonstationary or $1/f$ type. To the best of our knowledge, there is only one work in the literature [9] that solves this problem optimally using a multiscale Wiener filter in wavelet domain. As an alternative, the proposed Kalman filter can be used for the estimation and prediction of the transmitted $1/f$ signal from the observation data in a recursive fashion where no extra steps of wavelet filtering is needed.

7. CONCLUSION

In this paper, we have developed continuous time state space representation and an optimal state estimation algorithm using Kalman filtering for self-similar processes. Beginning from the most general and mathematically tractable dynamical representation such as Euler-Cauchy type differential equation definition of $1/f$ processes, the dynamics of the states are represented with respect to the multiplicative group derivatives where the memory is captured in infinitesimal scalings of time.

Using this state space representation, we formulate the continuous time Kalman filter to estimate or predict the self-similar or $1/f$ data. Although the algorithm appears to be in the same form of LTI systems, the major difference is once again in the memory content or the dynamics of the estimated state and the error covariance matrix which is appropriate to capture the self-similar nature of the statistics.

This work can be extended to several further research areas. Here we assumed that the state space system parameters i.e. **A**, **B**, **C** and **D** are available. However this may not be possible in some real time applications, such as in network traffic or in fading channels in communication networks. Therefore, a generalized Kalman filtering technique that estimates and updates the unknown system matrices can further be investigated. This framework can be extended and tested for 2D self-similar signals such as deblurring of textured images.

8. REFERENCES

[1] Birsen Yazici, R. Kashyap, "A Cl Processes for 1/f Phenomena," *IEEE Trans. on Signal Processing*, Vol.45, No. 2, 1997.

[2] B. Mandelbrot, "Some Noises with 1/f Spectrum: A Brass of Second-Order Stationary Self-Similar Processes for 1/f Phenomena," *IEEE Trans. on Signal Processing*, Vol.45, No. 2, 1997.

[3] M. S. Keshner, "1/f Noise," *Proc. IEEE*, Vol. 70, pp. 212-218, 1982.

[4] A. Van Der Ziel, "Unified Presentation of 1/f Noise in Electronic Devices: Fundamental 1/f Noise," *Proc. of IEEE*, Vol. 76, No. 3, 1988.

[5] G. W. Wornell, *Signal Processing with Fractals: A Wavelet-Based Approach*, Prentice Hall, 1996.

[6] M. M. Daniel, A. S. Willsky, " The Modeling and Estimation of Statistically Self-Similar Processes in a Multiresolution Framework," *IEEE Trans. on Information Theory*, Vol. 45, No. 3, pp. 955-970, 1999.

[7] W. E. Leland, M. S. Taqqu, W. Willinger, D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Trans. On Networking*, Vol. 2, No. 1, pp. 1-15, 1994.

[8] R. Rao, S. Lee, "Self-Similar Traffic Characterization Through Linear Scale Invariant System Models," *Proc. of IEEE International Conf. on Personal Wireless Communication*, pp. 138-142, 2000.

[9] B.S. Chen, Y.C. Chung, D.F. Huang, "Optimal Time-Frequency Deconvolution Filter Design for Nonstationary Signal Transmission Through a Fading Channel: AF Filter Bank Approach," *IEEE Trans. On Signal Processing*, Vol. 46, No. 12, pp. 3220-3234, 1998.

[10] G.S. Chirikjian, A.B. Kyatkin, *Engineering applications of noncommutative harmonic analysis*, CRC Press, 2001.

[11] A. Gelb, *Applied Optimal Estimation*, The M.I.T. Press, 1974.

[12] T. Kailath, *Linear Systems*, Prentice Hall, 1980.

[13] W. F. Arnold, A. J. Laub, "Generalized Eigenproblem Algorithms and Software for Algebraic Riccati Equations," *Proc. of IEEE*, Vol. 72, No. 12, pp. 1746-1754, 1984.
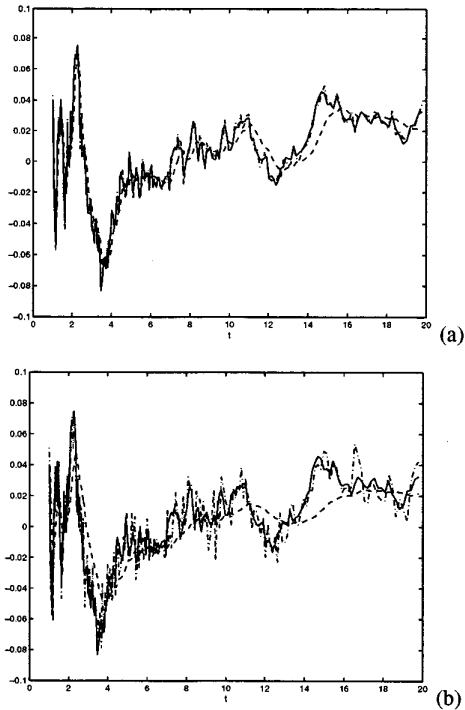
Figure 1: The input signal, $x(t)$ (solid line), the observed signal, $y(t)$ (dash-dot line) and the predicted signal, $\hat{x}(t)$ (dashed line) for a)input $SNR = 20$dB b) input $SNR = 10$dB.