A Data Concealing Technique with Random Noise Disturbance and A Restoring Technique for the Concealed Data by Stochastic Process Estimation

Tomohiro Fujii*        Masao Hirokawa†‡

October 9, 2019

Abstract

We propose a technique to conceal data on a physical layer by disturbing them with some random noises, and moreover, a technique to restore the concealed data to the original ones by using the stochastic process estimation. Our concealing-restoring system manages the data on the physical layer from the data link layer. In addition to these proposals, we show the simulation result and some applications of our concealing-restoring technique.

1 Introduction

Micro-device technology in the near future realizes the remote control of microprocessor chips in several things such as household electric appliances, information-processing equipments, and even brain-computer interface from the outside through the wireless or the so-called IoT (i.e., Internet of Things). Moreover, it enables the automatic operation of such things with that remote control. They are going to infiltrate society and play several important roles in every area of society. We then have to establish the security for them. In particular, we have to stem the hacking of the remote control and the wiretapping of the data of communication.

We are interested in a data concealing technique with disturbance on a physical layer and a restoring technique for those concealed data. Here, the physical layer is the lowest layer of the open systems interconnection (OSI) [13] (see Fig.1). OSI is a reference model to grasp and analyze how data are sent and received over a computation or communication network. Some methods using disturbance have been presented to conceal data for storage and communication. For instance, chaotic cryptology [6, 9, 16, 28] uses chaos to make the disturbance. The method using cryptographic hash functions for the disturbance has lately been gaining a practical position [7, 17, 18, 26]. Particularly, there have been some endeavors for the concealing technique on physical layers: the chaos multiple-input multiple-output [12, 20, 21, 22, 30]. Meanwhile, it is noteworthy that there have been many studies on the secured telecommunication using noises [8, 11, 19, 27, 29]. For the signal put on a carrier wave and sent from an antenna, we send some noises from interference antennas; we have the signal interfering with them and make it an interference wave.

We take interest in how to conceal data on a physical layer using some random noise disturbances and how to restore those concealed data applying a stochastic filtering theory to maintain the safety of data over a proper period of time, which is different from the interference wave method. Thus, our concealing-restoring system should be installed on the data link layer above the physical layer (see Fig[1]). The idea of the concealing-restoring system that we propose in this paper is primarily originated in keeping security over a
necessary period for the data processing on the physical layer of our developing quantum-sensing device. This device detects and handles some ultimate personal information. Since we are required to remove several noises in the device, we make our concealing-restoring system coexist with the denoising system of the device. As some applications deriving therefrom, we establish a mathematical technique for concealing data by the disturbance with randomness of the noises, and moreover, a mathematical technique for restoring the concealed data by the stochastic process estimation. In addition to these establishments, we show the simulation result and some applications for the two techniques. The idea on our method to conceal data comes from an image of the scene when we conceal a treasure map, and it is so simple as follows:

c1) we plaster over the treasure map at random and make it messy;
c2) we repeat c1 and plaster it over repeatedly.

In this paper, we mathematically realize c1 and c2, and make their implementation. In addition to c1 and c2, we can consider that

c3) we tear the muddled map by c1 and c2, and split it into several pieces

though we do not make its implementation in this paper.

We are planning that we use the concealed data for saving them in a memory or for sending them for telecommunication. We expect to use our methods in the situation where the physical layer is under restrictions in the implementation space due to a small consumed electric power, a small arithmetic capacity, a small line capacity, and a bad access environment, and moreover, in the situation where it is too harsh to make a remote maintenance of the physical layer, for example, in outerspace development or seafloor development. We hope to apply the implementation of our techniques, for instance, to the remote control of drones and devices on them, and to the security of some data sent from those devices.

2 Mathematical Set-Ups

We first explain the outline of how to make our concealing-restoring system for data $X_t, t \in \mathbb{R}$, which is given by a simultaneous equation system (SES). This SES consists of some stochastic differential equations (SDEs), linear equations, and a nonlinear equation (NLE). The data
X_t is input as the initial data of the SES. We prepare N functionals \( F_i, i = 1, 2, \cdots, N \), making the SDEs. We suppose that each form of the individual functional \( F_i \) is known only by those who conceal the original data \( X_t \) and restore their concealed data. We use the forms of the functionals as well as the composition of the SES for secret keys or common keys. We prepare 2N random noises \( W_{t,i}^j, j = 1, 2; i = 1, 2, \cdots, N \), for the SDEs, and a nonlinear bijection \( f \) for the NLE. The SDEs for processes \( X_t^i, i = 1, 2, \cdots, N \), and the NLE for the process \( X_t^{N+1} \) are used to introduce the noise disturbance in our concealing-restoring system. We also use the means, variances, and distributions of the random noises as well as the nonlinear bijection as secret keys. As shown below, we obtain \( X_t^i, i = 1, 2, \cdots, N \), and the NLE are made from the binary word \( a_0a_1\cdots a_n \) in the following.

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \) with \( \bar{a}_i \) for each \( i = 0, 1, \cdots, n \). For \( n + 1 \) bits, \( a_0, a_1, \cdots, a_n \in \{0, 1\} \), we concatenate them and make a word \( a_0a_1\cdots a_n \). We employ the following linear interpolation as a simple digital-analogue (D/A) transformation. We first define \( X_t^i \) by

\[
X_t^i = \begin{cases} 
  +1 & \text{if } a_i = 1, \\
  -1 & \text{if } a_i = 0,
\end{cases} \quad i = 0, 1, \cdots, n.
\]

We here explain how to make the data \( X_t^i \) from binary data. We use the low/high-signal for the binary data in this paper though there are many other ways. Thus, we represent ‘low’ by 0 and ‘high’ by 1. For \( n + 1 \) bits, \( a_0, a_1, \cdots, a_n \in \{0, 1\} \), we concatenate them and make a word \( a_0a_1\cdots a_n \). We employ the following linear interpolation as a simple digital-analogue (D/A) transformation. We first define \( X_t^i \) by

\[
X_t^i = \begin{cases} 
  +1 & \text{if } a_i = 1, \\
  -1 & \text{if } a_i = 0,
\end{cases} \quad i = 0, 1, \cdots, n.
\]

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \), \( 0 \leq t \leq n \). When the data \( X_t^i \) are made from the binary word \( a_0a_1\cdots a_n \), we call \( X_t^i \) a binary pulse for the word \( a_0a_1\cdots a_n \). As for the restoration of the word, we use the simple analogue-digital (A/D) transformation to seek the character \( \bar{a}_i \in \{0, 1\} \) for each \( i = 0, 1, \cdots, n \), and make a word \( \bar{a}_0\bar{a}_1\cdots \bar{a}_n \) for the original word \( a_0a_1\cdots a_n \) in the following.

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \), \( 0 \leq t \leq n \). When the data \( X_t^i \) are made from the binary word \( a_0a_1\cdots a_n \), we call \( X_t^i \) a binary pulse for the word \( a_0a_1\cdots a_n \). As for the restoration of the word, we use the simple analogue-digital (A/D) transformation to seek the character \( \bar{a}_i \in \{0, 1\} \) for each \( i = 0, 1, \cdots, n \), and make a word \( \bar{a}_0\bar{a}_1\cdots \bar{a}_n \) for the original word \( a_0a_1\cdots a_n \) in the following.

We here explain how to make the data \( X_t^i \) from binary data. We use the low/high-signal for the binary data in this paper though there are many other ways. Thus, we represent ‘low’ by 0 and ‘high’ by 1. For \( n + 1 \) bits, \( a_0, a_1, \cdots, a_n \in \{0, 1\} \), we concatenate them and make a word \( a_0a_1\cdots a_n \). We employ the following linear interpolation as a simple digital-analogue (D/A) transformation. We first define \( X_t^i \) by

\[
X_t^i = \begin{cases} 
  +1 & \text{if } a_i = 1, \\
  -1 & \text{if } a_i = 0,
\end{cases} \quad i = 0, 1, \cdots, n.
\]

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \), \( 0 \leq t \leq n \). When the data \( X_t^i \) are made from the binary word \( a_0a_1\cdots a_n \), we call \( X_t^i \) a binary pulse for the word \( a_0a_1\cdots a_n \). As for the restoration of the word, we use the simple analogue-digital (A/D) transformation to seek the character \( \bar{a}_i \in \{0, 1\} \) for each \( i = 0, 1, \cdots, n \), and make a word \( \bar{a}_0\bar{a}_1\cdots \bar{a}_n \) for the original word \( a_0a_1\cdots a_n \) in the following.

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \), \( 0 \leq t \leq n \). When the data \( X_t^i \) are made from the binary word \( a_0a_1\cdots a_n \), we call \( X_t^i \) a binary pulse for the word \( a_0a_1\cdots a_n \). As for the restoration of the word, we use the simple analogue-digital (A/D) transformation to seek the character \( \bar{a}_i \in \{0, 1\} \) for each \( i = 0, 1, \cdots, n \), and make a word \( \bar{a}_0\bar{a}_1\cdots \bar{a}_n \) for the original word \( a_0a_1\cdots a_n \) in the following.

We connect \( X_t^i \) and \( X_t^{i+1} \) with a straight line for each \( i = 0, 1, \cdots, n - 1 \), and we have polygonal line \( X_t^i \), \( 0 \leq t \leq n \). When the data \( X_t^i \) are made from the binary word \( a_0a_1\cdots a_n \), we call \( X_t^i \) a binary pulse for the word \( a_0a_1\cdots a_n \). As for the restoration of the word, we use the simple analogue-digital (A/D) transformation to seek the character \( \bar{a}_i \in \{0, 1\} \) for each \( i = 0, 1, \cdots, n \), and make a word \( \bar{a}_0\bar{a}_1\cdots \bar{a}_n \) for the original word \( a_0a_1\cdots a_n \) in the following.
Eqs. (2.1) and (2.3) are the mathematical realization of $c_1$. The repetition of Eq. (2.1) from $i = 1$ to $i = N$ with the help of Eq. (2.2) is for the mathematical realization of $c_2$. We can mathematically realize $c_3$ by taking numbers $r_\ell$, $\ell = 1, 2, \cdots, M$, with $\sum_{\ell=1}^{M} r_\ell = 0$, and defining
\[
U_\ell^i = \frac{1}{M} \left( U_\ell^i + r_\ell W_\ell^j \right), \quad \ell = 1, 2, \cdots, M,
\]
where $i \neq j$. Then, we can split the data $U_\ell^i$ into the data $U_\ell^\ell$, $\ell = 1, 2, \cdots, M$. For instance, we generate a random number $r$ with $r \neq 0$, and set $r_1$ and $r_2$ as $r_1 = r$ and $r_2 = -r$ for $M = 2$. From the split data, $U_\ell^\ell$, $\ell = 1, 2, \cdots, M$, we can restore the data $U_\ell^\ell$ to the data $U_\ell^i$ and $U_\ell^j$ by
\[
U_\ell^i = \sum_{i=1}^{M} U_\ell^i \quad \text{and} \quad U_\ell^j = r^{-1}(MU_\ell^\ell - U_\ell^i)
\]
for an $\ell$ satisfying $r_\ell \neq 0$. We can use the sequence, $r_1, r_2, \cdots, r_M$, as a secret or common key.

We note that the last stochastic process appearing in Eq. (2.3) has the form,
\[
X_t^{N+1} = c_1 \cdots c_N X_t + \sum_{i=1}^{N-1} \left( \prod_{j=i+1}^{N} c_j \right) W_t^{2,i} + W_t^{2,N}.
\]

**2.1 How to Conceal Data**

We take the original signal $X_t$ as initial data,
\[
X_1^1 = X_t.
\]
Inputting it into Eq. (2.1) with the noise $W_t^{1,1}$, we conceal it by the SDE,
\[
F_1(X_1^1, X_t^1, U_1^1, W_t^{1,1}) = 0.
\]

We seek $U_1^1$ in the above and obtain a concealed data $U_1^1$. By Eq. (2.2),
\[
X_2^2 = c_1 X_1^1 + W_t^{2,1},
\]
we have data $X_2^2$ for the next step. These data $X_2^2$ consist of the superposition (i.e., linear combination) of $X_1^1$ and $W_t^{2,1}$, and thus, there is a possibility that a wiretapper remove the noise $W_t^{2,1}$ and wiretap $X_1^1$. Thus, to improve the security with another noise-disturbance, we have the same procedure again. We input the data $X_2^2$ into Eq. (2.1) with the noise $W_t^{1,2}$,
\[
F_2(X_2^2, X_2^1, U_2^2, W_t^{1,2}) = 0.
\]

We then obtain the concealed data $U_2^2$. We repeat the same procedures, and obtain the concealed data, $U_1^1, U_2^2, \cdots, U_t^N$, and hide the data, $X_1^1, X_2^2, \cdots, X_t^N$.

At last, inputting the concealed data $X_t^N$ into Eq. (2.2) and get the data $X_t^{N+1}$. We input this into Eq. (2.3) and hide it. We then obtain the last concealed data $U_t^{N+1}$. In this way, the sequence of the concealed data, $U_1^1, U_2^2, \cdots, U_t^N, U_t^{N+1}$, are created.

In the case where the original data are digital, and they give the binary pulse $X_t$, the concealed data, $U_t^i$, $i = 1, 2, \cdots, N, N + 1$, merely become analogue data. So, a wiretapper has to know A/D transformation to obtain the original digital data as getting the concealed data. Therefore, the D/A and A/D transformations play an important role for the concealing-restoring system for some digital data. We can also use them as secret or common key.

**2.2 How to Restore Data**

Since the nonlinear function $f$ is bijective, we can restore the concealed data $U_t^{N+1}$ to the data $X_t^{N+1}$ by
\[
X_t^{N+1} = f^{-1}(U_t^{N+1}).
\]
In the light of the stochastic filtering theory, Eqs. (2.1) and (2.2) are the state equation and the observation equation, respectively. That is, they make the system form the noise-filtering. Inputting the above $X_t^{N+1}$ into Eq. (2.2), and the concealed data $U_t^N$ into Eq. (2.1), we have simultaneous equations to seek the data $X_t^N$,

$$F_N(X_t^N, \dot{X}_t^N, U_t^N, W_t^{1,N}) = 0,$$

$$X_t^{N+1} = c^N X_t^N + W_t^{2,N}.$$  

Since we cannot completely restore the noises to the original ones, $W_t^{1,N}$ and $W_t^{2,N}$, we estimate the stochastic process $X_t^N$ with the help of a proper stochastic filtering theory to remove the random noises. We then obtain the estimated data $\hat{X}_t^N$.

Inputting the estimated data $\hat{X}_t^N$ into the slot of $X_t^N$ of Eq. (2.2), and the concealed data $U_t^{N-1}$ into Eq. (2.1), we reach simultaneous equations to seek the data $X_t^{N-1}$,

$$F_{N-1}(X_t^{N-1}, \dot{X}_t^{N-1}, U_t^{N-1}, W_t^{1,N-1}) = 0,$$

$$\hat{X}_t^N = c^{N-1} X_t^{N-1} + W_t^{2,N-1}.$$  

In the same way as in the above, the stochastic filtering theory gives us the next estimated data $\hat{X}_t^N$. We repeat this procedure, and obtain the estimated data, $\hat{X}_t^N, \hat{X}_t^{N-1}, \cdots, \hat{X}_t^2, \hat{X}_t^1$, by turns, and we pick up the last estimate $\hat{X}_t^1$. This is the restoration $\hat{X}_t$ of the original data $X_t$.

### 3 Example of Functionals and Simulation

As for how to determine each functional, $F_i$, $i = 1, 2, \cdots, N$, any definition of it is fine so long as a noise-filtering theory is established for the system with $F_i$. To restore the concealed data, $U_t^1, U_t^2, \cdots, U_t^N, U_t^{N+1}$, generally speaking, we have to know the concrete forms of the functionals, and the noise-filtering theory. Therefore, we hide the both for securing the original data.

Though it is actually supposed to be in secret, we give one of examples of the concrete definition of the functionals in this paper. We determine functions $A^i(t), v^i(t)$ and non-zero constants $b_{iu}^i, b_{i}^i$ in secret. Here $v^i(t)$ can be a random noise. For instance, we often makes $v^i(t)$ by the linear interpolation based on normal random numbers. Namely, we first assign a normal random number with $N(0, \sigma_v^2)$ to $v^i(t)$ for each $i$ and $k$, and then, connect them by linear interpolation. Here $N(0, \sigma_v^2)$ means the normal distribution whose mean and standard deviation are respectively 0 and $\sigma_v$. We give each functional $F_i$ such that it makes a SDE,

$$dX_i^t = (A^i(t) - 1) X_i^t dt + b_{i}^i U_t^i dt + b_{iu}^i v^i(t) dt - b_{iu}^i dB_t^i,$$  \hspace{1cm} (3.1)

for each $i = 1, 2, \cdots, N$. That is,

$$\dot{X}_i^t = (A^i(t) - 1) X_i^t + b_{iu}^i U_t^i + b_{i}^i v^i(t) - b_{iu}^i W_t^{1,i}.$$  \hspace{1cm} (3.2)

Here $W_t^{1,i}$ and $W_t^{2,i}$ are Gaussian white noises whose mean $m^{i,i}$ and variance $V^{i,i}$ are respectively 0 and $(\sigma_{j}^i)^2$. $B_t^i$ is the Brownian motion given by $W_t^{1,i} = dB_t^i$. We assume that the noises $W_t^{1,i}$ and $W_t^{2,i}$ are independent for each $i = 1, 2, \cdots, N$, but the noises $W_t^{2,i}$, $i = 1, 2, \cdots, N$, are not always independent. Thus, in the case where they are not independent, the linear combination of white noises appearing in Eq. (2.1) is not always white noise.

We regard the functions $A^i(t)$, the constants $b_{iu}^i, b_{i}^i$, and the mean $m^{i,i}$ and variance $V^{i,i} = (\sigma_{j}^i)^2$ of the white noises as secrete keys which are known only by the administrator of our concealing-restoring system. We use functions $v^i(t)$ as common keys which are used for both concealing and restoring the data. Since Eqs. (3.1) and (2.2) respectively play the individual roles of the state equation and observation equation in the stochastic filtering theory, we employ the linear Kalman filtering theory [4, 10, 14, 15] to seek the restoration $\hat{X}_t$.  

1
3.1 Discrete Version of Kalman Filtering

We seek the restoration \( \hat{X}_t \) by numerical analysis. Thus, we discretize Eq.\((2.1)\); we approximate the differential by the forward difference,

\[
\frac{dX^i_t}{dt} \approx \frac{X^i_{t+\Delta t} - X^i_t}{\Delta t},
\]

for \( t = k\Delta t \) with \( k = 0, 1, 2, \ldots \). Employing \( \Delta t \) as a unit, the differential is approximated as \( dX^i_t/dt = X^i_{k+1} - X^i_k \) for \( k = 1, 2, \ldots \), and therefore, Eq.\((2.1)\) is discretized as

\[
X^i_{k+1} = A^i_k X^i_k + b^i_k U^i_k + b^i_k v^i_k - b^i_k W^i_k
\]

for each \( i = 1, 2, \ldots, N \). Here we respectively denote \( A^i(k) \) and \( v^i(k) \) by \( A^i_k \) and by \( v^i_k \). With this discretization, Eqs.\((2.2)\) and \((2.3)\) respectively become

\[
X^{i+1}_k = c^i X^i_k + W^{2,i}_k, \quad i = 1, 2, \ldots, N,
\]

\[
U^{N+1}_k = f(X^{N+1}_k).
\]

In addition to the concealed data \( U^{N+1}_k \) given by Eq.\((3.3)\), using Eq.\((3.2)\) we give the other concealed data \( U^i_k, i = 1, 2, \ldots, N \), by

\[
U^i_k = \frac{1}{b^i_k} \{ X^i_{k+1} - A^i_k X^i_k - b^i_k v^i_k \} + W^{i+1}_k.
\]

The concealed data are actually created by a computer with Alg.1.

**Algorithm 1 Concealing Data \( X_t \)**

1. Determine secret keys, \( A^i_k, b^i_k, b^i_k, c^i; m^{j,i}, \sigma_j^i \).
2. Determine a common key, \( v^i_k \).
3. Define white noises, \( W^{j,i}_k, j = 1, 2; i = 1, 2, \ldots, N \), with the individual mean \( m^{j,i} \) and variance \( (\sigma_j^i)^2 \).
4. Determine \( N \), how many SDEs you want.
5. Determine \( n \), how many data you handle.
6. for \( k = 0, 1, \ldots, n \) do
   7. Set \( X^i_k := X^i_k \)
   8. end for
9. for \( i = 1, 2, \ldots, N \) do
   10. for \( k = 0, 1, \ldots, n \) do
        11. Set \( U^i_k := (b^i_k)^{-1} \{ X^i_{k+1} - A^i_k X^i_k - b^i_k v^i_k \} + W^{i+1}_k \)
        12. Set \( X^{i+1}_k := c^i X^i_k + W^{2,i}_k \)
   13. end for
   14. end for
15. end for
16. Set \( U^{N+1}_k := f(X^{N+1}_k) \)

Conversely, we can estimate the data, \( X^N_k, X^{N-1}_k, \ldots, X^1_k \), from the concealed data, \( U^N_k, U^{N-1}_k, \ldots, U^1_k \). The linear Kalman filtering theory \([3][10][14][15]\) says that we can make an algorithm to obtain the estimated data, \( \hat{X}^N_k, \hat{X}^{N-1}_k, \ldots, \hat{X}^1_k \), in the following. We denote the priori estimate by \( \hat{X}^{-1}_k \), the variance by \( P^i_k \), the priori variance by \( P^{-1}_k \), and the Kalman gain by \( g^i_k \) for each \( i = N, N - 1, \ldots, 1 \). We repeat the procedure consisting of ‘Prediction Step’ and ‘Filtering Step’ from \( i = N \) to \( i = 1 \). We note that in the Kalman filtering theory the estimate \( \hat{X}_k \) represents an optimal estimate, and is called the posteriori estimate.
Before the Kalman filtering, the concealed data $\hat{X}_{k}^{N+1}$ is obtained as

$$\hat{X}_{k}^{N+1} = f^{-1}(U_{k}^{N+1}), \quad k = 0, 1, \ldots, n.$$  

‘Prediction Step’ and ‘Filtering Step’ of the Kalman filtering are as follows:

**Prediction Step:**

$$\hat{X}_{k}^{-i} = A_{k}^{i} \hat{X}_{k-1}^{i} + b_{u}^{i}U_{k-1}^{i} + b^{i}v_{k-1}^{i},$$

$$P_{k}^{-i} = (A_{k}^{i})^{2}P_{k-1}^{i} + (\sigma_{1}^{i})^{2}(b_{u}^{i})^{2}.$$  

**Filtering Step:**

$$g_{k} = \frac{c^{i}P_{k}^{-i}}{(c^{i})^{2}P_{k}^{-i} + (\sigma_{2}^{i})^{2}},$$

$$\hat{X}_{k}^{i} = \hat{X}_{k}^{-i} + g_{k}(\hat{X}_{k+1}^{i} - c^{i}\hat{X}_{k}^{-i}),$$

$$P_{k}^{i} = (1 - c^{i}g_{k})P_{k}^{-i},$$

$\hat{X}_{0}^{-i}, P_{0}^{i}, g_{0}^{N+1}$, and $\hat{X}_{0}^{i}$.

The stochastic-process estimation consisting of the prediction and filtering steps is done by a computer with Alg.2

**Algorithm 2** Restoration from Concealed Data $U_{k}^{i}$, $i = 1, 2, \ldots, N, N + 1$

Get secret keys, $A_{k}^{i}, b_{u}^{i}, b^{i}, c^{i}, m^{i}, \sigma_{j}^{i}$,

Get the common key, $v_{k}^{i}$.

Obtain the concealed data, $U_{k}^{i}$.

for $k = 0, 1, \ldots, n$

Set $\hat{X}_{k}^{N+1} := f^{-1}(U_{k}^{N+1})$

end for

for $i = N, N - 1, \ldots, 0$

Determine initial values, $\hat{X}_{0}^{-i}, P_{0}^{i}, g_{0}^{i}$, $\hat{X}_{k}^{i}$.

for $k = 1, 2, \ldots, n$

Set $\hat{X}_{k}^{i} := A_{k}^{i}\hat{X}_{k-1}^{i} + b_{u}^{i}U_{k-1}^{i} + b^{i}v_{k-1}^{i}$

Set $P_{k}^{-i} := (A_{k}^{i})^{2}P_{k-1}^{i} + (\sigma_{1}^{i})^{2}(b_{u}^{i})^{2}$

Set $g_{k}^{i} := c^{i}P_{k}^{-i}((c^{i})^{2}P_{k}^{-i} + (\sigma_{2}^{i})^{2})^{-1}$

Set $\hat{X}_{k}^{i} := \hat{X}_{k}^{-i} + g_{k}^{i}(\hat{X}_{k+1}^{i} - c^{i}\hat{X}_{k}^{-i})$

Set $P_{k}^{i} := (1 - c^{i}g_{k}^{i})P_{k}^{-i}$

end for

end for

for $k = 0, 1, \ldots, n$

Set $\hat{X}_{k} := \hat{X}_{k}^{i}$

end for

**3.2 Simulation of Concealing and Restoring Data on Physical Layer**

In our simulation of concealing and restoring data on physical layer, we employ the message digest [3][23][24][25] to check the coincidence of the original word $a_{0}a_{1} \cdots a_{n}$ and its restored word $\tilde{a}_{0}\tilde{a}_{1} \cdots \tilde{a}_{n}$ though the message digest works on upper layers. Moreover, we can use
the message digest to detect any falsification of the concealed data. We take the original word \(a_0a_1\cdots a_n\) as a message, and then, produce its digest. We also produce the digest for the restored word \(\hat{a}_0\hat{a}_1\cdots \hat{a}_n\). Comparing hash values of the two digests, we can make the check of the coincidence and the detection of the falsification at the same time. They should be done on one out of layers between Layer 3 and Layer 7. In our simulation, we employ SHA-256 to make the hash values [3].

To make the estimation in the simulation, we employ the linear Kalman filtering theory [15] under the following conditions. We make Eqs.(2.1)-(2.3) for \(N = 2\) with \(A'(t) = 0.1\) (constant function), \(b^i = 1\), \(b'^i = 1\), and \(c^i = 1\) for each \(i = 1, 2\). We define the common key \(v^i(t)\) by the linear interpolation based on a normal random number with \(N(0, 1^2)\). We assume that the mean of white noises are all 0. The standard deviation of the white noise \(W^1_t\) is \(\sigma_1^1 = 0.1\), and that of the white noise \(W^2_t\) is \(\sigma_1^2 = 1\). The length of the word \(a_0a_1\cdots a_n\) is 100, and therefore, \(n = 99\).

Our original word \(a_1a_2\cdots a_{99}\) is given by Eq.(3.5). We here note that we remove the character \(a_0\) because we cannot estimate the first bit in our concealing-restoring system.

\[
\begin{array}{c}
00001100101100001011111111100100110100111011111111111100100110101011111111111110011000101100011011001001001101111100110111001101110111010011110010110101100011110011080
\end{array}
\]  

Then, we get its binary pulse \(X_t\) as in Fig.2. The hash value of the digest made from the original word (3.5) is

\[
979bca61579e002c9097c78088740e9fdaf21535d6a5c5876bd8623a86185292. \quad (3.6)
\]

We make the concealed data, \(U^1_t\) and \(U^2_t\), by Eq.(3.4) with the help of the linear equation given in Eq.(2.2). We finally make the concealed data \(U^3_t\) using the nonlinear equation given in Eq.(2.3) with \(f(\xi) = \xi^3\). Their graphs are in Figs.3 & 4. Following the Kalman filtering theory, we remove the white noises, and estimate the binary pulse \(X_t\). Then, we obtain the restoration \(\hat{X}_t\) as in Fig.5. Let us take 0 as the threshold. Then, we obtain the restored word \(\hat{a}_1\hat{a}_2\cdots \hat{a}_{99}\) and the hash value of its digest made from the restoration \(\hat{X}_t\), and they are the same as Eqs.(3.5) and (3.6), respectively. We note that the graphs in Figs.3 & 4 say that the concealed data, \(U^1_t\), \(U^2_t\), and \(U^3_t\), are merely analogue data. If a wiretapper becomes aware that the concealed data are for digital ones and knows our A/D transformation in some way, then the wiretapper gets a binary word from the concealed data as follows:

\[
\begin{array}{c}
0011101100011110110001111000110100110111111111001101100011110011010110101101100011110011080
\end{array}
\]

Figure 2: The binary pulse \(X_t\) transformed from the original word (3.5).
Figure 3: The concealed data, $U^1_t$ (left) and $U^2_t$ (right), for the binary pulse $X_t$ in Fig.2.

Figure 4: The concealed data $U^3_t$ for the binary pulse $X_t$ in Fig.2.

Figure 5: The restoration $\hat{X}_t$ for the binary pulse $X_t$ in Fig.2.
for $U_1^t$, 
\[
001101100111011010110000100110011011101000100110110101
01011001010011111010011110101010010011001111010110110110110
\]
for $U_2^t$, and 
\[
1000000000011100110001100100011001110010001101110010001100110100
000010110111110101100010001000001001100111101011011001
\]
for $U_3^t$. Here, since the wiretapper does not know that we remove the first bit, every concealed data $U_i^t$ make the word consisting of 100 characters.

In Fig. 6 we show the comparison of the original binary pulse $X_t$, its restoration $\tilde{X}_t$, and the concealed data $U_i^t, i = 1, 2, 3$.

![Figure 6](image)

Figure 6: $X_t$ (Fig. 2) and $\tilde{X}_t$ (Fig. 5) from the above of the left 2 graphs. $U_1^t$ (Fig. 3), $U_2^t$ (Fig. 3), and $U_3^t$ (Fig. 4) from the above of the right 3 graphs. Here $t \in [0, 99]$.

4 Application to Data on Physical Layer and Presentation Layer

4.1 Binary Data of Pictorial Image

We now apply the technology of our mathematical method to the binary data of a pictorial image. We use digital data of a pictorial image in the ORL Database of Faces, an archive of AT&T Laboratories Cambridge [2]. The data have the greyscale value of 256 gradations (8bit/pixel). We set our parameters as $A = A^t = 0.1$, $b = b^t = 1$, $b_u = b_u^t = 1$, $c = c^t = 1$, $\sigma_1 = \sigma_1^t = 0.1$, and $\sigma_2 = \sigma_2^t = 1$. We determine the common key $v^t(t)$ in the same way as in §3.2 with $\sigma_v = 2$. The original pictorial image and its binary pulse $X_t$ are obtained as in Fig. 7. Here, the upper bound of $t$ is $92 \times 112 = 10304$ and $t$ runs over $[0, 10304]$. We obtain the concealed data, $U_1^t$ and $U_2^t$, by Eq. (3.4) as in Fig. 8 and the concealed data $U_3^t$ by Eq. (3.3) as in Fig. 9. The restoration $\tilde{X}_t$ and the restored pictorial image from it are in Fig. 10.

Since the concealed data, $U_1^t$, $U_2^t$, and $U_3^t$, are analogue as in Figs 8 & 9, a wiretapper has to know our A/D transformation, and moreover, our transformation from the digital
Figure 7: The original pictorial image (left) with the digital data, and its binary pulse $X_t$ (right) only for $t \in [0, 200]$.

Figure 8: The concealed data, $U^1_t$ (left) and $U^2_t$ (right), for the binary pulse $X_t$ in Fig.7 Here $t \in [0, 200]$ only.

Figure 9: The concealed data $U^3_t$ for the binary pulse $X_t$ in Fig.7 Here $t \in [0, 200]$ only.
data to a pictorial image at least as well as our concealing method with Eqs. (2.1)-(2.3). The
latter transformation should be done on upper layers. We now assume that the wiretapper
can know the transformations. Then, the pictorial images of the concealed data, \( U_1^t, U_2^t, \) and \( U_3^t \), are in Fig. 11.

As for the role of the common key \( v_i(t) \), comparing Fig. 12 with Fig. 11, we can realize the
effect of the variance of the common key \( v^i(t) \) and the nonlinear function \( f(\xi) \). The variance
of the common key \( v^i(t) \) is smaller in Fig. 12 than it is in Fig. 11 that is, \( (\sigma_v)^2 = 4 \) for Fig. 11
and \( (\sigma_v)^2 = 1 \) for Fig. 12, though other parameters for Fig. 12 are the same as for Fig. 11.
The contour of the face in the pictorial image of \( U_1^t \) in Fig. 12 stands out more clearly than in
Fig. 11. Meanwhile, the nonlinearity conceals the contour as in the pictorial image of \( U_3^t \) in
Fig. 12.

In Fig. 13 we show the comparison of the original binary pulse \( X_t \), its restoration \( \hat{X}_t \), and
the concealed data \( U_i^t, i = 1, 2, 3 \).

### 4.2 Analogue Data of Pictorial Image

We use analogue data of a pictorial image in the Olivetti faces database [11], where the data
of pictorial images are transformed to analogue data from the original ones in the ORL
Database of Faces, an archive of AT&T Laboratories Cambridge [2]. The data have the
greyscale value of 256 gradations (8bit/pixel). Our parameters are \( A = A_i = 0.1, b = b^i = 1, \)
\( b_u = b_u^i = 1, c = c^i = 1, \sigma_1 = \sigma_1^i = 0.1, \) and \( \sigma_2 = \sigma_2^i = 1 \) again. We also use the common
key \( v^i(t) \) in the same way as in (3.2) with \( \sigma_v = 2 \). The original analogue data \( X_t \) and their
pictorial image are in Fig. 14. Here, the upper bound of \( t \) is \( 64 \times 64 = 4096 \) and \( t \) runs.
Figure 12: From the left, pictorial images of the concealed data, \( U_1^t, U_2^t \) in Fig.8 and \( U_3^t \) in Fig.9 for the binary pulse \( X_t \) in Fig.7. Here \( (\sigma_v)^2 = 1.000000 \).

Figure 13: \( X_t \) (Fig.7) and \( \hat{X}_t \) (Fig.10) from the above of the left 2 graphs. \( U_1^t \) (Fig.8), \( U_2^t \) (Fig.8), and \( U_3^t \) (Fig.9) from the above of the right 3 graphs. Here \( t \in [0, 200] \) only.
over [0, 4096]. The concealed data, $U_1^t$ and $U_2^t$, defined by Eq. (3.4) are in Fig. 15 and the
concealed data $U_3^t$ defined by Eq. (3.3) is in Fig. 16. We can restore the pictorial image with
the restoration $\hat{X}_t$ as in Fig. 17. If a wiretapper becomes aware our method to make a pictorial
image from analogue data, then the wiretapper gets pictorial images from the concealed data
$U_i^t$, $i = 1, 2, 3$, as in Fig. 18.

In Fig. 19 we show the comparison of the original binary pulse $X_t$, its restoration $\hat{X}_t$, and
the concealed data $U_i^t$, $i = 1, 2, 3$.

5 Conclusion and Future Work

We have proposed a mathematical technique for concealing data on the physical layer of
the OSI reference model by using random noise disturbance, and moreover, a mathematical
technique for restoring the concealed data by using the stochastic process estimation. In
this concealing-restoring system, the functionals determining SDEs play a role of secret or
common keys, and moreover, the proper noise-filtering theory forms a nucleus to restore the
concealed data. In addition, we have showed the simulation result for the data on physical
layer and some applications of the two techniques to the pictorial images. We have opened one
of examples of the functionals. Then, we have showed how to conceal the data by using the
noise-disturbance, and have demonstrated how to restore the data by removing the noises.
Here, the significant point to be emphasized is that any composition of the SES and any form
of the individual functional will do so long as a proper noise-filtering method is established for
Figure 16: The concealed data $U^3_t$ for the analogue data $X_t$, $t \in [0, 200] \subset [0, 4096]$, in Fig. 14.

Figure 17: The restoration $\hat{X}_t$ (right) for the analogue data $X_t$ in Fig. 14 only for $t \in [0, 200]$, and the pictorial image (left) of $\hat{X}_t$.

Figure 18: From the left, pictorial images of the concealed data, $U^1_t$ (Fig. 15), $U^2_t$ (Fig. 15), and $U^3_t$ (Fig. 16).
Although the state space determined by Eq. (3.1) is linear, Gaussian, and we used the linear Kalman filtering theory in §3, we can make it more general: nonlinear, non-Gaussian state space. Then, we should employ the nonlinear Kalman filtering theory and particle filtering theory [4]. In fact, we already checked that the particle filtering theory works.

We have used the scalar-valued processes, and thus, prepared just one common key for one SDE. We can prepare some common keys for one SDE by using the vector-valued processes. Although we have employed the message digest to make the check of the coincidence of the binary word and the detection of the falsification at the same time, we are now developing a method with low complexity so that we can make them for data on the physical layer.

According to our several experiments including the concrete examples in §4.4, we think that the nonlinearity enhances the noise-disturbance. For instance, the pictorial images in Fig. 20 are the case \( N = 1 \). Comparing the pictorial images of \( U^2_t \) and \( X^2_t = f^{-1}(U^2_t) \) in Fig. 21, we can say that the enhancement of noise-disturbance appears with color. We will study the roles of several parameters.

Figure 20: From the left, the original pictorial image, the individual pictorial images of the concealed data \( U^1_t \) and \( U^2_t \), and the pictorial image of the restored data. The original pictorial image is a bitmap image, and the parameter \( t \) of the original data \( X_t \) runs over \([0, 90123\text{byte}]\).
Figure 21: Comparison between the pictorial images of $U_t^2$ with nonlinearity (left) and $X_t^2 = f^{-1}(U_t^2)$ without nonlinearity (right).

Acknowledgments

This work is partially based on Fujii’s bachelor thesis at Hiroshima University in March, 2019. For useful comments and discussion, the authors thank the following: Kirill Morozov (University of North Texas), Shuichi Ohno (Hiroshima University), Kouichi Sakurai (Kyushu University), Takeshi Takagi (Hiroshima University), and Tatsuya Tomaru (HITACHI Ltd.).

References

[1] The Olivetti Faces Database. https://scikit-learn.org/0.19/datasets/olivetti_faces.html.
[2] The ORL Database of Faces. https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.
[3] Secure Hash Standard (SHS). Federal Information Processing Standards Publication, 2015.
[4] A. Bain and D. Crisan. Fundamentals of Stochastic Filtering. Stochastic Modelling and Applied Probability. Springer, 2009.
[5] S. binti Suhaili and T. Watanabe. High-throughput message digest (md5) design and simulation-based power estimation using unfolding transformation. Journal of Signal Processing, 21(6):233–238, November 2017.
[6] K. M. Cuomo and A. V. Oppenheim. Synchronization of lorenz-based chaotic circuits with applications to communications. IEEE Trans. on Circuit and Syst., 40:626–633, 1993.
[7] I. Damgård. A design principle for hash functions. In In Advances in Cryptology CRYPTO ’89 Proceedings, volume 435 of Lecture Notes in Computer Science, pages 416–427. Springer-Verlag, 1989.
[8] S. Goel and R. Negi. Guaranteeing secrecy using artificial noise. IEEE Trans. Wireless Commun., 7(6):2180–2189, 2008.
[9] G. Grassi and S. Mascolo. A system theory approach for designing cryptosystems based on hyperchaos. IEEE Trans. on Circuit and Systems -I: Fundamental Theory and Application, 46(9):1135–1138, 1999.
[10] M. S. Grewal and A. P. Andrews. Kalman Filtering. Theory and Practice Using MATLAB. Wiley, 2015.
[11] A. Hero. Secure space-time communication. IEEE Trans. Inf. Theory, 49(12):3235–3249, 2003.
[12] Y. Masuda K. Ito and E. Okamoto. A chaos mimo-based polar concatenation code for secure channel coding. IEEE, Jan. 2019.

[13] B. N. Kain and A. K. Agrawala. Open Systems Interconnection: Its Architecture and Protocols. McGraw-Hill, 1992.

[14] G. Kallianpur. Stochastic Filtering Theory. Applications of Mathematics 13. Springer-Verlag, 1980.

[15] R. E. Kalman. A new approach to linear filtering and prediction problems. Transactions of the ASME - Journal of Basic Engineering (Series D), 82:35–45, 1960.

[16] H. Lenug and J. Lam. Design of demodulator for the chaotic modulation communication system. Trans. Circuit Syst., 44:262–267, 1997.

[17] R. C. Merkle. Secrecy, authentication, and public key systems. Ph.d. thesis, Stanford University, 1979.

[18] R. C. Merkle. A certified digital signature. In In Advances in Cryptology - CRYPTO ’89 Proceedings, volume 435 of Lecture Notes in Computer Science, pages 218–238. Springer-Verlag, 1989.

[19] A. Mukherjee and A. L. Swindlehurst. Robust beam-forming for security in mimo wiretap channels with imperfect csi. IEEE Trans. Signal Process., 59(1):351–361, 2011.

[20] E. Okamoto. A chaos mimo transmission scheme for secure communications on physical layer. IEEE, Jul. 2011.

[21] E. Okamoto and Y. Inaba. A chaos mimo transmission scheme using turbo principle for secure channel-coded transmission. IEICE Transactions on Communications, E98.B(8):1482–1491, August 2015.

[22] E. Okamoto and Y. Iwanami. A trellis-coded chaotic modulation scheme. volume 11, pages 5010–5015, Jun. 2006.

[23] R. Rivest. The MD4 message digest algorithm, volume 37 of Lecture Notes in Computer Science. Springer-Verlag, 1991.

[24] R. Rivest. The MD5 Message Digest Algorithm. MIT and RSA Data Security, Inc., April 1992.

[25] R. Rivest. The MD5 Message Digest Algorithm. MIT and RSA Data Security, Inc., April 1992.

[26] Bruce Schneier. Applied Cryptography: Protocols, Algorithms and Source Code in C. Wiley, 2015.

[27] A. L. Swindlehurst. Fixed sinr solution for the mimo wiretap channel. Proc. IEEE International Conf. Acoustics, Speech and Signal Processing, pages 2437–2440, 2009.

[28] C. W. Wu and L. O. Chua. A simple way to synchronize chaotic systems with applications to secure communication systems. Int. J. Bifurcation Chaos, 3(6):1619–1627, 1993.

[29] A. D. Wyner. The wire tap channel. The Bell Syst. Tech. J., 54(8):1355–1387, 1975.

[30] G. Zheng. Secure communication based on multi-input multi-output. Chaotic System with Large Message Amplitude, Chaos, Solitons & Fractals, 41(3):1510–1517, 2009.