PAPER

Statistical-Mechanical Analysis of Adaptive Volterra Filter with the LMS Algorithm

Kimiko MOTONAKA[†a)], ***Member***, **Tomoya KOSEKI**[††], ***Nonmember***, **Yoshinobu KAJIKAWA**[†], ***and*** **Seiji MIYOSHI**[†], ***Senior Members***

SUMMARY The Volterra filter is one of the digital filters that can describe nonlinearity. In this paper, we analyze the dynamic behaviors of an adaptive signal-processing system including the Volterra filter by a statistical-mechanical method. On the basis of the self-averaging property that holds when the tapped delay line is assumed to be infinitely long, we derive simultaneous differential equations in a deterministic and closed form, which describe the behaviors of macroscopic variables. We obtain the exact solution by solving the equations analytically. In addition, the validity of the theory derived is confirmed by comparison with numerical simulations.

key words: *Volterra filter, adaptive filter, statistical-mechanical informatics, LMS algorithm*

1. Introduction

Adaptive signal-processing techniques have been used in various fields, such as information communication and acoustic systems [1], [2]. There have been many studies on such techniques using a linear digital filter and also on theoretical analysis [3]–[5]. On the other hand, there are analytical techniques using a statistical-mechanical method [6]. The statistical-mechanical method is producing significant results in many fields, such as associative memory models [7], [8], error-correcting codes [9], wireless communications [10], image processing [11], statistical learning [12] and so forth. Statistical-mechanical analysis assumes a large limit of system size, and its merit is that it can describe the universal feature of problems macroscopically. This technique has been applied to the analysis of the behaviors of digital filters. As an example, a linear digital filter for active noise control has been analyzed by a statistical-mechanical method [13]–[17]. On the other hand, in an actual environment, the target system is often nonlinear; thus, a digital filter that can describe this nonlinearity is desirable. There are many nonlinear filters i.e. order statistics filters [18] or morphological filters [19], [20] and so on. The Volterra filter is one of such digital filters that can describe nonlinearity [21]. It can effectively describe weak nonlinearity with no hysteresis and has been applied to the modeling of loudspeaker systems and other applications [22]–[25]. The second-order Volterra filter has been analyzed in some previous studiess [26]–[28]. Takahama et al. [26] analyzed the error signal of the second-order Volterra filter by decomposing it into the coefficient errors of diagonal elements and off-diagonal elements and transforming the Volterra filter into the first-order recursive filter. Chao and Inomata [27] analyzed the geometric property of the error surface and the convergence property of the adaptive algorithm by eigenvalue analysis of the input autocorrelation matrix of the second-order Volterra filter. Koh and Powers [28] analyzed the steady state of the second-order Volterra filter using the cross-bicorrelation function of the input and output of the unknown system. However, analyzing the third- or higher-order Volterra filter is also important, especially if we consider its actual applications. Moreover, analyzing the Lth-order, that is, the generalized-order Volterra filter is significant from the viewpoint of theoretical analysis. Therefore, in this paper, we analyze the dynamic behavior of an adaptive signal-processing system including the Lth-order Volterra filter by a statistical-mechanical method and obtain the exact solution regarding the behavior of the mean square error (MSE). In the analysis of this paper, unlike in Refs. [26]–[28], the MSE is derived explicitly as a function of time, the order of the Volterra filter, the input signal size, and the step size, as we shall see later. Thus, we can obtain insights into the effects of these parameters on the behavior of the system.

Although many update algorithms have been proposed for the adaptive Volterra filter, here, we analyze the case of using the least-mean-square (LMS) algorithm [1], [2], [29]. In this paper, on the basis of the self-averaging property [6] that holds when the tapped delay line is assumed to be infinitely long, we derive simultaneous differential equations in a deterministic and closed form, which describe the behaviors of the macroscopic variables. We obtain the exact solution by solving the equations analytically. Then, we verify the behavior of the Volterra filter while changing the step size and background noise. Finally, we confirm that the analytically obtained dynamic behavior of the mean square error (learning curves) is in good agreement with numerical simulations.

In this paper, the overview of the Volterra filter is described in Sect. 2, and the analytical model is shown in Sect. 3. In Sects. 4 and 5, the derivation of the differential equations for the second-order and Lth-order Volterra

Manuscript received January 22, 2021.
Manuscript revised March 9, 2021.
Manuscript publicized June 1, 2021.

[†]The authors are with the Department of Electrical, Electronic and Information Engineering, Faculty of Engineering Science, Kansai University, Suita-shi, 564-8680 Japan.

[††]The author is with JB Toll System. Co. Ltd., Kobe-shi, 651-0084 Japan.

a) E-mail: motonaka@kansai-u.ac.jp

DOI: 10.1587/transfun.2021EAP1013

filter, respectively, and the exact solution obtained by solving the equations analytically are described. In Sect. 6, some results are shown to confirm the appropriateness of the derived solution by comparing with the computer experiments. Finally, conclusions are presented in Sect. 7.

2. Volterra Filter

The Volterra filter is one of the digital filters that can describe nonlinearity, and it uses the Volterra kernel of the Volterra series expansion. The discrete Volterra series expansion up to the Lth-order Volterra kernel with tap length N is defined as

$$\begin{aligned} y(n) = &\sum_{k_1=0}^{N-1} h_{k_1} x(n-k_1) \\ &+ \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} h_{k_1,k_2} x(n-k_1)x(n-k_2) \\ &+ \dots \\ &+ \sum_{k_1=0}^{N-1} \dots \sum_{k_L=0}^{N-1} h_{k_1,\dots,k_L} \prod_{k\in\{k_1,\dots,k_L\}} x(n-k). \end{aligned} \quad (1)$$

Here, $x(n)$ and $y(n)$ are the input and output signals in time step n, respectively. $h_{k_1,\dots,k_l}$ is the l-th Volterra kernel. In the adaptive Volterra filter, each Volterra kernel is updated. Note that in the case of $L = 1$, it is equivalent to a simple linear FIR filter. Without any loss of generality, one can assume that the Volterra kernels are symmetric, i.e., $h_{k_1,\dots,k_l}$ is left unchanged for any of the possible $l!$ permutations of the indices $k_1,\dots,k_l$ [21]. For example, in the case of the second-order and third-order Volterra kernels,

$$\begin{aligned} h_{k_1,k_2} &= h_{k_2,k_1}, \\ h_{k_1,k_2,k_3} &= h_{k_1,k_3,k_2} = h_{k_2,k_1,k_3} \\ &= h_{k_2,k_3,k_1} = h_{k_3,k_1,k_2} = h_{k_3,k_2,k_1} \end{aligned}$$

holds.

3. Analytical Model

The Volterra filter applied to adaptive signal processing is called the adaptive Volterra filter. Various methods, which are the same as those used with a simple linear adaptive filter, such as the gradient method and the recursive least-squares (RLS) method, can be used to update the adaptive Volterra filter, and we analyze the case in which the LMS algorithm, which is one of the gradient methods, is used for updating. Figure 1 shows a block diagram of the adaptive system using the LMS algorithm. In Fig. 1, P and H denote the unknown system and the adaptive filter, respectively. Here, an error signal $e(n)$ is calculated by adding the background noise $\xi(n)$ to the difference between $d(n)$ and $u(n)$, which are the output signals of P and H, respectively. Therefore, we obtain

$$e(n) = d(n) - u(n) + \xi(n). \quad (2)$$

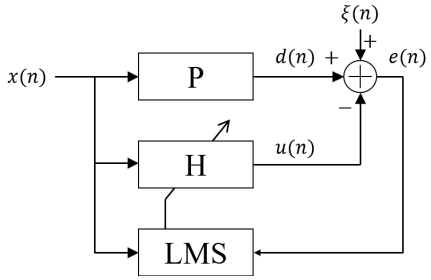

Fig. 1 Block diagram of the adaptive system using LMS algorithm.

Here, the background noise $\xi(n)$ is generated independently from a distribution with a mean of zero and a variance of σ_ξ^2.

3.1 Second-Order Volterra Filter [21], [30]

To proceed step by step, we describe the case in which P and H are constructed from the second-order Volterra kernels with tap length N. Their coefficient matrices are $\boldsymbol{p} = \{p_{k_1,k_2}\}$, $\boldsymbol{h}(n) = \{h_{k_1,k_2}(n)\}$, $k_1, k_2 = 0,\dots,N-1$. Here, to converge the first-order Volterra kernel, $O(N)$ updates are required. On the other hand, to converge the second-order Volterra kernel, $O(N^2)$ updates are required. Thus, if N is large, the convergence time of the second-order Volterra kernel is slower than that of the first-order Volterra kernel. In other words, the second-order Volterra kernel is dominant in the learning curve. Therefore, in the model of the second-order Volterra filter, the first-order Volterra kernel is omitted and the model includes only the second-order Volterra kernel. Each element p_{k_1,k_2}, $k_1 \le k_2$, is generated independently from a distribution with a mean of zero and a variance of unity. In addition, the Volterra kernel $\boldsymbol{p}$ is symmetric as described in Sect. 2. The initial matrix $\boldsymbol{h}(0)$ is set to be a zero matrix. The input signal $x(n)$ is generated independently from a distribution with

$$\langle x(n) \rangle = 0, \quad (3)$$

$$\left\langle x^2(n) \right\rangle = \sigma^2. \quad (4)$$

Here, the tap input vector $\boldsymbol{x}(n)$ in time step n is

$$\boldsymbol{x}(n) = [x(n), x(n-1), \cdots, x(n-N+1)]^T. \quad (5)$$

$d(n)$ and $u(n)$, which are the output signals of P and H, respectively, in time step n are

$$d(n) = \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} p_{k_1,k_2} x(n-k_1)x(n-k_2), \quad (6)$$

$$u(n) = \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} h_{k_1,k_2}(n) x(n-k_1)x(n-k_2). \quad (7)$$

The update formula of each element of the second-order adaptive Volterra filter using the LMS algorithm is

$$h_{k_1,k_2}(n+1) = h_{k_1,k_2}(n) + \mu e(n) x(n-k_1)x(n-k_2), \quad (8)$$

where μ denotes the step-size parameter [21].

3.2 Lth-Order Volterra Filter [21]

In this section, we describe the case in which P and H are constructed from the Lth-order Volterra kernels, $\boldsymbol{p} = \{p_{k_1,\ldots,k_L}\}$, $\boldsymbol{h}(n) = \{h_{k_1,\ldots,k_L}(n)\}$, $k_1,\ldots,k_L = 0,\ldots,N-1$. Here, to converge the lth-order Volterra kernel, $O(N^l)$ updates are required. Thus, as in the second-order case, the highest-order Volterra kernel converges slower than the lower-order Volterra kernels and is dominant in the learning curve. Therefore, as with the model of the second-order Volterra filter, the Volterra kernels that are lower than the Lth-order Volterra kernel are omitted, and the model includes only the Lth-order Volterra kernel. Each element $p_{k_1,k_2,\ldots,k_L}, k_1 \le k_2 \le \ldots \le k_L$ is generated independently from a distribution with a mean of zero and a variance of unity. In addition, the Volterra kernels $\boldsymbol{p}$ are symmetric as described in Sect. 2. The input signal $x(n)$ is generated independently from a distribution with a mean of zero and a variance of σ^2. Here, the tap input vector $\boldsymbol{x}(n)$ in time step n is the same as that in the case of the second-order system and written as Eq. (5).

$d(n)$ and $u(n)$, which are the output signals of P and H, respectively, in time step n are

$$d(n) = \sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} p_{k_1,\ldots,k_L} \prod_{k\in\{k_1,\ldots,k_L\}} x(n-k), \quad (9)$$

$$u(n) = \sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} h_{k_1,\ldots,k_L}(n) \prod_{k\in\{k_1,\ldots,k_L\}} x(n-k). \quad (10)$$

The update formula of each element of the Lth-order adaptive Volterra filter using the LMS algorithm is

$$h_{k_1,\ldots,k_L}(n+1) = h_{k_1,\ldots,k_L}(n) + \mu e(n) \prod_{k\in\{k_1,\ldots,k_L\}} x(n-k), \quad (11)$$

where μ denotes the step-size parameter [21]. Note that this model becomes a linear filter when $L = 1$. Therefore, the analytical result in the case of $L = 1$ exactly agrees with the result reported by Ishibushi et al. [17].

4. Theory for the Second-Order Volterra Filter [30]

In this section, we theoretically analyze the behaviors of the second-order adaptive Volterra filter by using a statistical-mechanical method.

4.1 Mean Square Error (MSE)

The MSE of the model of the second-order Volterrra filter can be calculated using Eq. (2) as follows:

$$\begin{aligned}\left\langle e^2(n)\right\rangle &= \left\langle (d(n)-u(n)+\xi(n))^2\right\rangle \\ &= \left\langle d^2(n)\right\rangle + \left\langle u^2(n)\right\rangle + \left\langle \xi^2(n)\right\rangle - 2\left\langle d(n)u(n)\right\rangle \\ &\quad + 2\left\langle d(n)\xi(n)\right\rangle - 2\left\langle u(n)\xi(n)\right\rangle \\ &= \left\langle d^2(n)\right\rangle + \left\langle u^2(n)\right\rangle - 2\left\langle d(n)u(n)\right\rangle + \sigma_\xi^2. \end{aligned} \quad (12)$$

In this paper, $\langle\cdot\rangle$ denotes the expectation with respect to the tap input vector $\boldsymbol{x}(n)$. Note that the background noise $\xi(n)$ is independent of the other stochastic variables.

Next, we focus on each term of Eq. (12). From Eq. (6), we obtain

$$\begin{aligned}\left\langle d^2(n)\right\rangle &= \left\langle \left(\sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} p_{k_1,k_2}x(n-k_1)x(n-k_2)\right)^2\right\rangle \\ &= \left\langle \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1}\sum_{k_1'=0}^{N-1}\sum_{k_2'=0}^{N-1} p_{k_1,k_2}p_{k_1',k_2'} \right. \\ &\qquad \left. x(n-k_1)x(n-k_2)x(n-k_1')x(n-k_2')\right\rangle. \end{aligned} \quad (13)$$

The right-hand side of Eq. (13) can be divided into the following four cases and others.

$$k_1 = k_2 = k_1' = k_2' \quad (14)$$

$$k_1 = k_1',\ k_2 = k_2',\ k_1 \ne k_2 \quad (15)$$

$$k_1 = k_2,\ k_1' = k_2',\ k_1 \ne k_1' \quad (16)$$

$$k_1 = k_2',\ k_2 = k_1',\ k_1 \ne k_2 \quad (17)$$

Note that we need not consider cases other than the above four cases because their expectations are zero owing to the products of independent components. The means of the products of the input signals corresponding to Eqs. (14)–(17) are as follows, respectively:

$$\left\langle x^4(n-k_1)\right\rangle = O\left(\sigma^4\right), \quad (18)$$

$$\left\langle x^2(n-k_1)x^2(n-k_2)\right\rangle = \sigma^4, \quad (19)$$

$$\left\langle x^2(n-k_1)x^2(n-k_1')\right\rangle = \sigma^4, \quad (20)$$

$$\left\langle x^2(n-k_1)x^2(n-k_2)\right\rangle = \sigma^4. \quad (21)$$

The right-hand side of Eq. (18) depends on the distribution that generates $x(n)$. If $x(n)$ is generated from the Gaussian distribution, the right-hand side of Eq. (18) is $3\sigma^4$ because the kurtosis of the Gaussian distribution is three. Now, we can rewrite Eq. (13) as

$$\begin{aligned}\left\langle d^2(n)\right\rangle &= O\left(\frac{r^2}{N^2}\right)\sum_{k_1=0}^{N-1} p_{k_1,k_1}^2 \\ &\quad + \frac{r^2}{N^2}\sum_{k_1=0}^{N-1}\sum_{\substack{k_2=0\\k_2\ne k_1}}^{N-1} p_{k_1,k_2}^2 \\ &\quad + \frac{r^2}{N^2}\sum_{k_1=0}^{N-1}\sum_{\substack{k_1'=0\\k_1'\ne k_1}}^{N-1} p_{k_1,k_1}p_{k_1',k_1'} \end{aligned}$$

$$\text{Summing } N^2dt \text{ equations} \quad \begin{cases} N^2R(n+1) & = N^2R(n) & +\mu e(n)d(n) \\ N^2R(n+2) & = N^2R(n+1) & +\mu e(n+1)d(n+1) \\ \vdots & \vdots & \vdots \\ N^2R(n+N^2dt) & = N^2R(n+N^2dt-1) & +\mu e(n+N^2dt-1)d(n+N^2dt-1) \end{cases} \tag{31}$$

$$N^2R(n+N^2dt) = N^2R(n) + \mu \sum_{i=0}^{N^2dt-1} e(n+i)d(n+i)$$

$$+ \frac{r^2}{N^2} \sum_{k_1=0}^{N-1} \sum_{\substack{k_2=0 \\ k_2 \neq k_1}}^{N-1} p_{k_1,k_2} p_{k_2,k_1}. \tag{22}$$

Here, $r = N\sigma^2$. Note that when we set $r = 1$, it corresponds to the Normalized LMS (NLMS) algorithm. Assuming $N \to \infty$ while keeping r constant in accordance with the statistical-mechanical method[†], the first and third terms become zero because the elements of $\boldsymbol{p}$ are assumed to be independently generated from a distribution with a mean of zero and a variance of unity, as described in Sect. 3. In addition, the Volterra kernel has symmetricity, that is, $p_{k_1,k_2} = p_{k_2,k_1}$. Therefore, $\left\langle d^2(n) \right\rangle$ can be rewritten as

$$\left\langle d^2(n) \right\rangle = \frac{2r^2}{N^2} \sum_{k_1=0}^{N-1} \sum_{\substack{k_2=0 \\ k_2 \neq k_1}}^{N-1} p_{k_1,k_2}^2 = 2r^2 \tag{23}$$

because p_{k_1,k_2} was generated independently from a distribution with a mean of zero and a variance of unity. We can also obtain $\left\langle u^2(n) \right\rangle$ and $\langle d(n)u(n) \rangle$ in Eq. (12) by the same procedure as that for $\left\langle d^2(n) \right\rangle$:

$$\left\langle u^2(n) \right\rangle = \frac{2r^2}{N^2} \sum_{k_1=0}^{N-1} \sum_{\substack{k_2=0 \\ k_2 \neq k_1}}^{N-1} h_{k_1,k_2}^2(n), \tag{24}$$

$$\langle d(n)u(n) \rangle = \frac{2r^2}{N^2} \sum_{k_1=0}^{N-1} \sum_{\substack{k_2=0 \\ k_2 \neq k_1}}^{N-1} p_{k_1,k_2} h_{k_1,k_2}(n). \tag{25}$$

Here, we assumed that there is little correlation between $\boldsymbol{x}(n)$ and $\boldsymbol{h}(n)$ [3]–[5].

Next, we introduce the macroscopic variables $R(n)$ and $Q(n)$ respectively defined as

$$R(n) = \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} p_{k_1,k_2} h_{k_1,k_2}(n), \tag{26}$$

$$Q(n) = \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} h_{k_1,k_2}^2(n). \tag{27}$$

Intuitively, $R(n)$ is the similarity between P and H and $Q(n)$ is the squared Frobenius norm of H. From Eqs. (12) and (23) – (27), we obtain

$$\left\langle e^2(n) \right\rangle = 2r^2 + 2r^2Q(n) - 4r^2R(n) + \sigma_\xi^2. \tag{28}$$

This equation shows that the mean square error (MSE) is a function of the macroscopic variables $R(n)$ and $Q(n)$.

4.2 Differential Equations of R and Q

Now, we derive the simultaneous differential equations that describe the dynamic behaviors of the macroscopic variables R and Q. First, we derive the differential equation of R. Multiplying both sides of Eq. (8) by p_{k_1,k_2} and summing them over k_1 and k_2, we obtain

$$\sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} p_{k_1,k_2} h_{k_1,k_2}(n+1) = \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} p_{k_1,k_2} h_{k_1,k_2}(n) + \mu e(n) \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} p_{k_1,k_2} x(n-k_1)x(n-k_2). \tag{29}$$

This can be rewritten as follows using Eqs. (6) and (26):

$$N^2R(n+1) = N^2R(n) + \mu e(n)d(n). \tag{30}$$

Note that the first terms on both sides of Eq. (30) are $O(N^2)$, although the second term on the right-hand side is $O(1)$. Thus, to change $R(n)$ by $O(1)$, $O(N^2)$ updates are required. Therefore, we use t, which is n normalized by N^2, as the time scale. By updating Eq. (30) N^2dt times in an infinitely small time dt, we can obtain N^2dt equations. Summing those N^2dt equations, the terms in both sides cancel each other as shown in Eq. (31). Therefore, we obtain

$$R(n+N^2dt) = R(n) + dt\mu \left\langle e(n)d(n) \right\rangle. \tag{32}$$

Assuming $N \to \infty$, the second term on the right-hand side of Eq. (32) is replaced by its mean[††]. We define the change in $R(t)$ by updating N^2dt times as $dR(t)$. From Eqs. (2), (23), (25), and (26), we obtain

[†]This is called the thermodynamic limit in statistical mechanics.

[††]This property is called self-averaging in statistical mechanics [6].

$$\begin{aligned}\frac{dR(t)}{dt} &= \mu \langle e(n)d(n)\rangle \\ &= \mu \langle (d(n) - u(n) + \xi(n))\, d(n)\rangle \\ &= \mu \left(\langle d^2(n)\rangle - \langle u(n)d(n)\rangle \right) \\ &= 2r^2\mu(1 - R(t)). \end{aligned} \tag{33}$$

Secondly, we derive the differential equation for Q. Squaring both sides of Eq. (8) and summing over k_1 and k_2, we obtain

$$\begin{aligned}&\sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} h^2_{k_1,k_2}(n+1) \\ &= \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} h^2_{k_1,k_2}(n) \\ &\quad + 2\mu e(n) \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} h_{k_1,k_2}(n)x(n-k_1)x(n-k_2) \\ &\quad + \mu^2 e^2(n) \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} x^2(n-k_1)x^2(n-k_2).\end{aligned}$$

This can be rewritten as follows using Eqs. (7) and (27):

$$\begin{aligned}N^2Q(n+1) &= N^2Q(n) + 2\mu e(n)u(n) \\ &\quad + \mu^2 e^2(n) \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{N-1} x^2(n-k_1)x^2(n-k_2).\end{aligned}$$

Similarly to the case of R, by updating N^2dt times in an infinitely small time dt and summing those N^2dt equations, we obtain

$$\begin{aligned}N^2Q(n+N^2dt) = N^2Q(n) &+ 2N^2dt\mu \langle e(n)u(n)\rangle \\ &+ N^2dtr^2\mu^2 \left\langle e^2(n)\right\rangle .\end{aligned} \tag{34}$$

Then, from Eqs. (2) and (24)–(27), we obtain

$$\begin{aligned}\frac{dQ(t)}{dt} &= 2\mu \langle e(n)u(n)\rangle + r^2\mu^2 \left\langle e^2(n)\right\rangle \\ &= 2\mu \left(\langle d(n)u(n)\rangle - \left\langle u^2(n)\right\rangle \right) \\ &\quad + r^2\mu^2 \left(\left\langle d^2(n)\right\rangle + \left\langle u^2(n)\right\rangle \right. \\ &\quad \left. + \left\langle \xi^2(n)\right\rangle - 2\left\langle d(n)u(n)\right\rangle \right) \\ &= 2\mu \left(2r^2R(t) - 2r^2Q(t) \right) \\ &\quad + r^2\mu^2 \left(2r^2 + 2r^2Q(t) + \sigma^2_\xi - 4r^2R(t) \right) \\ &= 4r^2\mu(R(t) - Q(t)) \\ &\quad + r^4\mu^2 \left(2 + 2Q(t) - 4R(t) + \frac{\sigma^2_\xi}{r^2} \right).\end{aligned} \tag{35}$$

4.3 Exact Solution of MSE

The derived differential equations for R and Q (Eqs. (33) and (35), respectively) can be solved analytically, and we obtain

$$R(t) = 1 - e^{-2r^2\mu t}, \tag{36}$$

$$\begin{aligned}Q(t) = 1 &+ \left(1 + \frac{\mu\sigma^2_\xi}{2r^2(r^2\mu - 2)} \right) e^{2r^2\mu(r^2\mu-2)t} \\ &- \frac{\mu\sigma^2_\xi}{2r^2(r^2\mu - 2)} - 2e^{-2r^2\mu t}.\end{aligned} \tag{37}$$

Substituting these equations into Eq. (28), we obtain the exact solution of the MSE as

$$\left\langle e^2(t)\right\rangle = \left(2r^2 + \frac{\mu\sigma^2_\xi}{r^2\mu - 2} \right) e^{2r^2\mu(r^2\mu-2)t} - \frac{2\sigma^2_\xi}{r^2\mu - 2}. \tag{38}$$

5. Theory for the Lth-Order Volterra Filter

In this section, we theoretically analyze the behaviors of the Lth-order adaptive Volterra filter by using a statistical-mechanical method. The flow of analysis and the basic technique are the same as those in the second-order case.

5.1 MSE

The MSE of the model of the Lth-order Volterra filter is also written as Eq. (12), that is,

$$\left\langle e^2(n)\right\rangle = \left\langle d^2(n)\right\rangle + \left\langle u^2(n)\right\rangle - 2\langle d(n)u(n)\rangle + \sigma^2_\xi. \tag{39}$$

In the case of the Lth-order Volterra filter, the first term of the right-hand side of Eq. (39) can be written as

$$\begin{aligned}&\left\langle d^2(n)\right\rangle \\ &= \left\langle \left(\sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} p_{k_1,\ldots,k_L} \prod_{k\in\{k_1,\ldots,k_L\}} x(n-k) \right)^2 \right\rangle \\ &= \left\langle \sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} \sum_{k'_1=0}^{N-1} \cdots \sum_{k'_L=0}^{N-1} p_{k_1,\ldots,k_L} \right. \\ &\qquad \left. p_{k'_1,\ldots,k'_L} \prod_{k\in\{k_1,\ldots,k_L\}} x(n-k) \prod_{k'\in\{k'_1,\ldots,k'_L\}} x(n-k') \right\rangle.\end{aligned} \tag{40}$$

Here, the Volterra kernels $\boldsymbol{p}$ are symmetric as described in Sect. 2. Additionally, as in Sect. 4, we introduce $r = N\sigma^2$. Note that when we set $r = 1$, it corresponds to the NLMS algorithm. Assuming $N \to \infty$ while keeping r constant in accordance with the statistical-mechanical method, we obtain

$$\begin{aligned}\left\langle d^2(n)\right\rangle &= L! \sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} p^2_{k_1,\ldots,k_L} \prod_{k\in\{k_1,\ldots,k_L\}} \left\langle x^2(n-k)\right\rangle \\ &= \frac{L!r^L}{N^L} \sum_{k_1=0}^{N-1} \cdots \sum_{k_L=0}^{N-1} p^2_{k_1,\ldots,k_L}\end{aligned}$$

$$= L!r^L \tag{41}$$

because $p_{k_1,...,k_L}$ was generated independently from a distribution with a mean of zero and a variance of unity, and the number of permutations of the set $\{k_1, ..., k_L\}$ is $L!$. Therefore, $2! = 2$ in Eq. (23) is replaced by $L!$ in Eq. (41). In addition, $\prod_{k\in\{k_1,...,k_L\}} \left\langle x^2(n-k)\right\rangle = r^L/N^L$ because $x(n)$ is generated independently from a distribution with a mean of zero and a variance of $\sigma^2 = r/N$. We can also obtain $\left\langle u^2(n)\right\rangle$ and $\langle d(n)u(n)\rangle$ in Eq. (39) by the same procedure as that in the case of $\left\langle d^2(n)\right\rangle$:

$$\left\langle u^2(n)\right\rangle = \frac{L!r^L}{N^L}\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} h^2_{k_1,...,k_L}(n), \tag{42}$$

$$\langle d(n)u(n)\rangle = \frac{L!r^L}{N^L}\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} p_{k_1,...,k_L} h_{k_1,...,k_L}(n). \tag{43}$$

Here, we assumed that there is little correlation between $\boldsymbol{x}(n)$ and $\boldsymbol{h}(n)$ [3]–[5].

Next, we introduce the macroscopic variables $R(n)$ and $Q(n)$ defined as follows:

$$R(n) = \frac{1}{N^L}\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} p_{k_1,...,k_L} h_{k_1,...,k_L}(n), \tag{44}$$

$$Q(n) = \frac{1}{N^L}\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} h^2_{k_1,...,k_L}(n). \tag{45}$$

From Eqs. (39) and (41)–(45), we obtain

$$\left\langle e^2(n)\right\rangle = L!r^L + L!r^L Q(n) - 2L!r^L R(n) + \sigma^2_\xi. \tag{46}$$

This equation shows that the MSE is a function of the macroscopic variables $R(n)$ and $Q(n)$.

5.2 Differential Equations of R and Q

Now, we derive the simultaneous differential equations that describe the dynamic behaviors of the macroscopic variables R and Q. First, we derive the differential equation of R. Multiplying both sides of the update formula Eq. (11) by $p_{k_1,...,k_L}$ and summing them over k_1,...,k_L, we obtain

$$\begin{aligned}
&\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} p_{k_1,...,k_L} h_{k_1,...,k_L}(n+1)\\
&= \sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} p_{k_1,...,k_L} h_{k_1,...,k_L}(n)\\
&+ \mu e(n)\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} p_{k_1,...,k_L}\prod_{k\in\{k_1,...,k_L\}} x(n-k).
\end{aligned}$$

This can be rewritten as follows using Eqs. (9) and (44),

$$N^L R(n+1) = N^L R(n) + \mu e(n)d(n). \tag{47}$$

Note that the first terms on both sides of Eq. (47) are $O(N^L)$, although the second term on the right-hand side is $O(1)$. Thus, to change $R(n)$ by $O(1)$, $O(N^L)$ updates are required. Therefore, we use t, which is n normalized by N^L, as the time scale. By updating Eq. (47) $N^L dt$ times in an infinitely small time dt, we can obtain $N^L dt$ equations, similarly to Eq. (31). Summing those $N^L dt$ equations as described in Sect. 4.2, we obtain

$$R(n + N^L dt) = R(n) + dt\mu\left\langle e(n)d(n)\right\rangle. \tag{48}$$

Assuming $N \to \infty$, the second term on the right-hand side of Eq. (48) is replaced by its mean based on the self-averaging property. We define the change in $R(n)$ by updating $N^L dt$ times as $dR(t)$. From Eqs. (2), (43), and (44), we obtain

$$\begin{aligned}
\frac{dR(t)}{dt} &= \mu\left\langle e(n)d(n)\right\rangle\\
&= \mu\langle(d(n) - u(n) + \xi(n))d(n)\rangle\\
&= \mu\left(\langle d^2(n)\rangle - \langle u(n)d(n)\rangle\right)\\
&= L!r^L\mu(1 - R(t)).
\end{aligned} \tag{49}$$

Secondly, we derive the differential equation for Q. Squaring both sides of Eq. (11) and summing over $k_1, ..., k_L$, we obtain

$$\begin{aligned}
&\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} h^2_{k_1,...,k_L}(n+1)\\
&= \sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} h^2_{k_1,...,k_L}(n)\\
&+ 2\mu e(n)\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1} h_{k_1,...,k_L}\prod_{k\in\{k_1,...,k_L\}} x(n-k)\\
&+ \mu^2 e^2(n)\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1}\prod_{k\in\{k_1,...,k_L\}} x^2(n-k).
\end{aligned}$$

This can be rewritten as follows using Eqs. (10) and (45):

$$\begin{aligned}
N^L Q(n+1) &= N^L Q(n) + 2\mu e(n)u(n)\\
&+ \mu^2 e^2(n)\left(\sum_{k_1=0}^{N-1}\cdots\sum_{k_L=0}^{N-1}\prod_{k\in\{k_1,...,k_L\}} x^2(n-k)\right).
\end{aligned}$$

Similarly to the case of R, by updating $N^L dt$ times in an infinitely small time dt and summing those $N^L dt$ equations, we obtain

$$\begin{aligned}
N^L Q(n + N^L dt) =& N^L Q(n) + 2N^L dt\mu\left\langle e(n)u(n)\right\rangle\\
&+ N^L dt\, r^L\mu^2\left\langle e^2(n)\right\rangle.
\end{aligned}$$

Then, from Eqs. (2) and (42)–(45), we obtain

$$\begin{aligned}
\frac{dQ(t)}{dt} &= 2\mu\left\langle e(n)u(n)\right\rangle + r^L\mu^2\left\langle e^2(n)\right\rangle\\
&= 2\mu\left\langle d(n)u(n) - u^2(n) - u(n)\xi(n)\right\rangle
\end{aligned}$$

$$
\begin{aligned}
&+ r^L\mu^2\left\langle d^2(n) + u^2(n) + \xi^2(n) - 2d(n)u(n)\right.\\
&\left.-2d(n)\xi(n) + 2u(n)\xi(n)\right\rangle\\
&= 2\mu\left(\langle d(n)u(n)\rangle - \left\langle u^2(n)\right\rangle\right)\\
&\quad + r^L\mu^2\left(\left\langle d^2(n)\right\rangle + \left\langle u^2(n)\right\rangle + \left\langle \xi^2(n)\right\rangle\right.\\
&\quad \left.-2\langle d(n)u(n)\rangle\right)\\
&= 2\mu\left(L!r^L R(t) - L!r^L Q(t)\right)\\
&\quad + r^L\mu^2\left(L!r^L + L!r^L Q(t) + \sigma_\xi^2 - 2L!r^L R(t)\right)\\
&= 2L!r^L\mu\left(R(t) - Q(t)\right)\\
&\quad + r^{2L}\mu^2\left(L! + L!Q(t) - 2L!R(t) + \frac{\sigma_\xi^2}{r^L}\right).
\end{aligned}
\tag{50}
$$

5.3 Exact Solution of MSE

The derived differential equations for R and Q (Eqs. (49) and (50), respectively) can be solved analytically, and we obtain

$$R(t) = 1 - e^{-L!r^L\mu t}, \tag{51}$$

$$
\begin{aligned}
Q(t) = 1 &+ \left(1 + \frac{\mu\sigma_\xi^2}{L!r^L(r^L\mu - 2)}\right)e^{L!r^L\mu(r^L\mu-2)t}\\
&- \frac{\mu\sigma_\xi^2}{L!r^L(r^L\mu - 2)} - 2e^{-L!r^L\mu t}.
\end{aligned}
\tag{52}
$$

By substituting these equations into Eq. (46), we obtain the exact solution of the MSE as

$$\left\langle e^2(n)\right\rangle = \left(L!r^L + \frac{\mu\sigma_\xi^2}{r^L\mu - 2}\right)e^{L!r^L\mu(r^L\mu-2)t} - \frac{2\sigma_\xi^2}{r^L\mu - 2}. \tag{53}$$

6. Results and Discussion

Using the statistical-mechanical method, we can discuss the universal property of the assumed system deterministically, regardless of its details, with a small number of macroscopic variables. As a result, it provides not only a highly accurate prediction of simulation results but also deep insight from the results obtained analytically. For example, from the exact solutions, Eqs. (38) and (53), we can obtain deep insight into the behavior of the MSE. For example, the necessary and sufficient condition for the convergence of the MSE is $0 < \mu < 2/r^L$. In this case, the steady-state MSE is $2\sigma_\xi^2/(2 - r^L\mu)$. Additionally, if the background noise does not exist, that is $\sigma_\xi^2 = 0$, the MSE is

$$
\begin{aligned}
\left\langle e^2(t)\right\rangle &= L!r^L e^{L!r^L\mu(r^L\mu-2)t}\\
&= L!r^L e^{-L!r^L(1-(r^L\mu-1)^2)t}.
\end{aligned}
$$

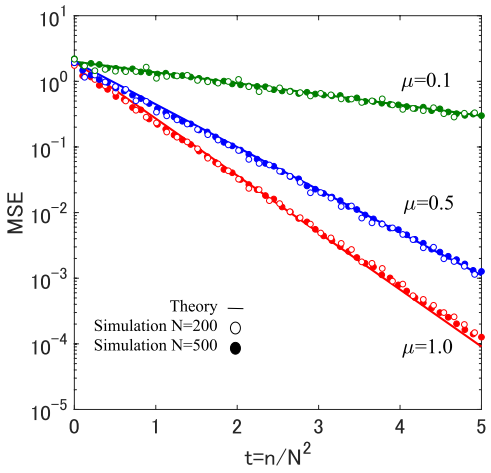

Fig. 2 Learning curves ($L = 2, r = 1, \sigma_\xi^2 = 0, \mu = 0.1, 0.5, 1.0$).

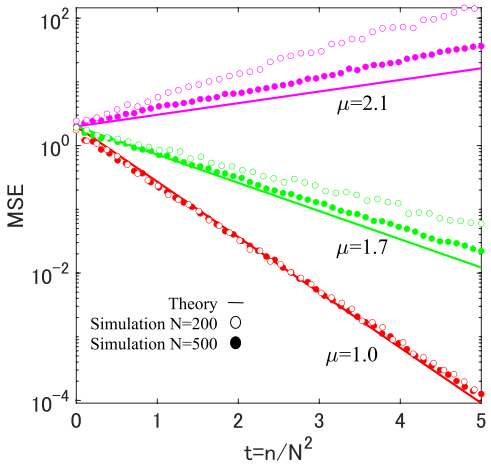

Fig. 3 Learning curves ($L = 2, r = 1, \sigma_\xi^2 = 0, \mu = 1.0, 1.7, 2.1$).

Thus, the MSE is minimum at $\mu = 1/r^L$ regardless of the values of t.

We compare the theoretical and simulation results in the case of the second-order Volterra filter. Note that, in all of the simulations in this study, r was set to $r = 1$. Figures 2 and 3 show the learning curves with $\sigma_\xi^2 = 0$, i.e., there is no background noise. In the figures, the solid lines denote the theoretical results and the symbols denote the results of numerical simulations. In the numerical simulations, the tap length was set to $N = 200, 500$ and the mean values of 100 trials are plotted. The open symbols show the results for $N = 200$ and the filled symbols show the results for $N = 500$. As shown in Figs. 2 and 3, the MSE becomes minimum when $\mu = 1.0$ regardless of the value of t. The systematic difference between the theoretical and numerical simulation results grows as μ increases. This is considered to be due to the finite-size effects [6] of the tap length N. Indeed, as shown in Figs. 2 and 3, the simulation results approach the theoretical results as the tap length N increases.

Figure 4 shows the learning curves with $\sigma_\xi^2 = 0.1$, i.e., background noise exists. In the figure, the solid lines

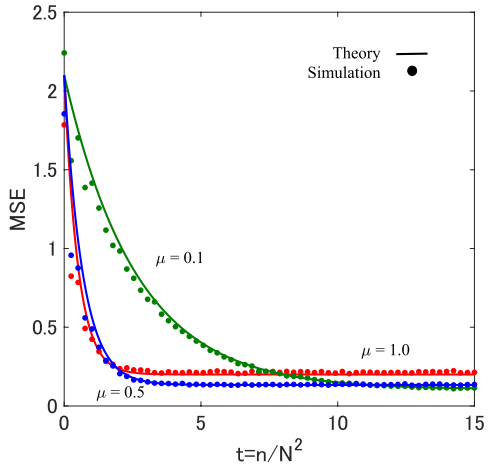

Fig. 4 Learning curves ($L = 2, r = 1, \sigma_\xi^2 = 0.1, \mu = 0.1, 0.5, 1.0$).

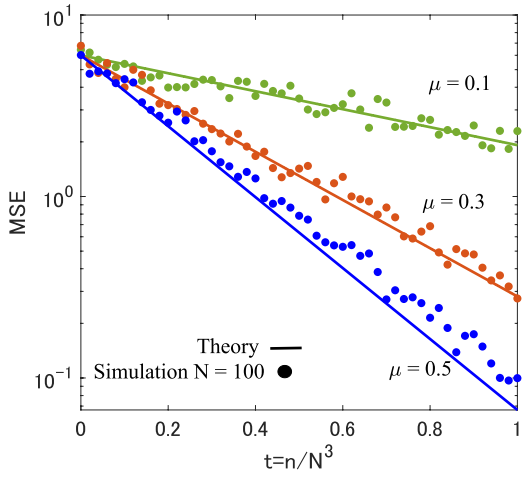

Fig. 5 Learning curves ($L = 3, r = 1, \sigma_\xi^2 = 0, \mu = 0.1, 0.3, 0.5$).

denote the theoretical results and the symbols denote the results of the numerical simulations. In the numerical simulations, the tap length was set to $N = 100$ and the mean values of 100 trials are plotted. From Fig. 4, we can confirm that the rate of decrease in the MSE in the early stage is greatest at $\mu = 1.0$, but after sufficient time has elapsed, the MSE becomes smaller as μ decreases; thus, the learning curves intersect. This result is predicted from the observation that Eqs. (38) and (53) converge to $2\sigma_\xi^2/(2 - r^L\mu)$ as $t \to \infty$.

Next, we compare the theory and the simulation results in the case of $L = 3$, which is the third-order Volterra filter. In the case of $L = 3$ and $r = 1$, from Eq. (53), the theoretical MSE is as follows:

$$\left\langle e^2(n)\right\rangle = \left(6 + \frac{\mu\sigma_\xi^2}{\mu - 2}\right) e^{6\mu(\mu-2)t} - \frac{2\sigma_\xi^2}{\mu - 2}.$$

Figure 5 shows the learning curves with $\sigma_\xi^2 = 0$, i.e., there is no background noise. The solid lines denote the theoretical results and the symbols denote the results of numerical simulations. In the case of $L = 3$, the tap length N and maximum time of simulation were set smaller than those in the case of $L = 2$ because the calculation cost of the simulation in the case of $L = 3$ is higher than that in the case of $L = 2$. In the numerical simulations, the tap length was set to $N = 100$ and the mean values of 100 trials are plotted. The systematic difference between the theoretical and numerical simulation results is larger than that in the case of $L = 2$ owing to the finite-size effects. However, the derived theoretical results are relatively in good agreement with the simulation results.

Figure 6 shows the learning curves with $\sigma_\xi^2 = 0.1$, i.e., background noise exists. In the figure, the solid lines denote the theoretical results and the symbols denote the results of the numerical simulations. In the numerical simulations, the tap length was set to $N = 50$ and the mean values of 100 trials are plotted. As shown in the figure, the behaviors are almost the same as that in the case of $L = 2$.

From the above results, it was confirmed that the exact

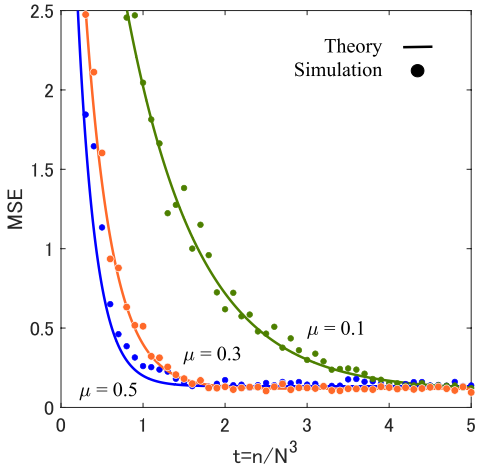

Fig. 6 Learning curves ($L = 3, r = 1, \sigma_\xi^2 = 0.1, \mu = 0.1, 0.3, 0.5$).

solution of the MSE derived in this paper is in good agreement with the results of the numerical simulations. The fact that the exact solution of the learning curves could be derived is significant from both standpoints of obtaining deep insight into Volterra filtering and its applications. In the statistical-mechanical analysis of the simple linear FIR filter, t, which is the number of updates n normalized by the tap length N, is used as the time scale [13]–[15]. However, in the analysis in this paper, it was necessary to normalize the number of updates n by the tap length N^L as the time scale t. This fact shows the essential slowness of the adaptation of the higher-order Volterra filter.

7. Conclusions

In this paper, we analyzed the dynamic behaviors of an adaptive signal processing system including the Volterra filter by a statistical-mechanical method. On the basis of the self-averaging property that can be true when the tapped delay line is assumed to be infinitely long, we derived simultaneous differential equations in a deterministic and closed form,

which describe the behaviors of the macroscopic variables, and we obtained the exact solution by solving them analytically. In addition, the validity of the derived theory was confirmed by comparison with numerical simulations.

Acknowledgments

This work was partially supported by JSPS KAKENHI Grant Numbers 17K06449, 18K13782, 18K11483, and 20K04494 and the Kansai University fund for the promotion and enhancement of education and research, 2018, "Development of system analysis and design methods integrating model and data premised on IOT technology".

References

[1] S. Haykin, Adaptive Filter Theory, 4th ed., Prentice Hall, Upper Saddle River, NJ, 2002.
[2] A.H. Sayed, Fundamentals of Adaptive Filtering, Wiley, Hoboken, NJ, 2003.
[3] M.H. Costa, J.C.M. Bermudez, and N.J. Bershad, "Stochastic analysis of the filtered-X LMS algorithm in systems with nonlinear secondary paths," IEEE Trans. Signal Process., vol.50, no.6, pp.1327–1342, 2002.
[4] O.J. Tobias, J.C.M. Bermudez, and N.J. Bershad, "Mean weight behavior of the filtered-X LMS algorithm," IEEE Trans. Signal Process., vol.48, no.4, pp.1061–1075, 2000.
[5] O.J. Tobias, J.C.M. Bermudez, R. Seara, and N.J. Bershad, "An improved model for the second moment of the filtered-X LMS algorithm," Proc. IEEE Adaptive Systems for Signal Processing, Communications, and Control Symp., pp.337–341, 2000.
[6] H. Nishimori, Statistical Physics of Spin Glasses and Information Processing: An Introduction, Oxford University Press, New York, 2001.
[7] M. Okada, "Notions of associative memory and sparse coding," Newral Networks, vol.9, no.8, pp.1429–1458, 1996.
[8] S. Miyoshi and M. Okada, "Storage capacity diverges with synaptic efficiency in an associative memory model with synaptic delay and pruning," IEEE Trans. Neural Netw., vol.15, no.5, pp.1215–1227, 2004.
[9] Y. Kabashima and D. Saad, "Statistical mechanics of error correcting codes," Europhys. Lett., vol.45, no.1, pp.97–103, 1999.
[10] T. Tanaka, "A statistical-mechanics approach to large-system analysis of CDMA multiuser detectors," IEEE Trans. Inf. Theory, vol.48, no.11, pp.2888–2910, 2002.
[11] K. Tanaka, "Statistical-mechanical approach to image processing," J. Phys. A: Math. Gen., vol.35, no.37, pp.R81–R150, 2002.
[12] A. Engel and C.V. Broeck, Statistical Mechanics of Learning, Cambridge University Press, Cambridge, 2001.
[13] S. Miyoshi and Y. Kajikawa, "Statistical-mechanical analysis of the FXLMS algorithm with nonwhite reference signals," Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), pp.5652–5656, 2013.
[14] S. Miyoshi and Y. Kajikawa, "Statistical-mechanical analysis of the FXLMS algorithm with actual primary path," Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), pp.3502–3506, 2015.
[15] S. Miyoshi and Y. Kajikawa, "Statistical-mechanics approach to the filtered-X LMS algorithm," Electron. Lett., vol.47, no.17, pp.997–999, 2011.
[16] S. Miyoshi and Y. Kajikawa, "Statistical-mechanics approach to theoretical analysis of the FXLMS algorithm," IEICE Trans. Fundamentals, vol.E101-A, no.12, pp.2419–2433, Dec. 2018.
[17] N. Ishibushi, Y. Kajikawa, and S. Miyoshi, "Statistical-mechanical analysis of LMS algorithm for time-varying unknown system," J. Phys, Soc. Jpn., vol.86, no.2, 024803, 2017.
[18] A. Bovik, T. Huang, and D. Munson, "A generalization of median filtering using linear combinations of order statistics," IEEE Trans. Acoust., Speech, Signal Process., vol.31, no.6, pp.1342–1350, 1983.
[19] P. Maragos and R. Schafer, "Morphological filters–Part I: Their set-theoretic analysis and relations to linear shift-invariant filters," IEEE Trans. Acoust., Speech, Signal Process., vol.35, no.8, pp.1153–1169, 1987.
[20] P. Maragos and R. Schafer, "Morphological filters–Part I: Their set-theoretic analysis and relations to linear shift-invariant filters," IEEE Trans. Acoust., Speech, Signal Process., vol.35, no.8, pp.1153–1169, 1987.
[21] V.J. Mathews, "Adaptive polynomial filters," IEEE Signal Process. Mag., vol.8, no.3, pp.10–26, 1991.
[22] T. Koh and E. Powers, "Second-order Volterra filtering and its application to nonlinear system identfication," IEEE Trans. Acoust., Speech, Signal Process., vol.33, no.6, pp.1445–1455, 1985.
[23] W. Frank, R. Roger, and U. Appel, "Loudspeaker nonlinearities–analysis and compensation," Comf. Record 26th Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA, vol.2, pp.756–760, 1992.
[24] W.A. Frank, "An efficient approximation to the quadratic Volterra filter and its application in real-time loudspeaker linearization," Signal Process., vol.45, no.1, pp.97–113, 1995.
[25] F.X.Y. Gao and W.M. Snelgrove, "Adaptive linearization of a loudspeaker," Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASP), vol.5, pp.3589–3592, 1991.
[26] Y. Takahama, Y. Kajikawa, and Y. Nomura, "A formulation of the convergence property for the second-order adaptive Volterra filter using NLMS algorithm," IEICE Trans. Fundamentals (Japanese edition), vol.J82-A, no.7, pp.944–953, July 1999.
[27] J. Chao, and A. Inomata, "An analysis of error surface and convergence of Volterra adaptive filters," IEICE Trans. Fundamentals (Japanese edition), vol.J82-A, no.6, pp.809–816, June 1999.
[28] T. Koh and E.J. Powers, "Second-order Volterra filtering and its application to nonlinear system identification," IEEE Trans. Acoust., Speech, Signal Process., vol.33, no.6, pp.1445–1455, 1985.
[29] B. Widrow and M.E. Hoff, Jr., "Adaptive switching circuits," IRE WESCON Conv. Rec., Pt. 4, pp.96–104, 1960.
[30] K. Motonaka, T. Katsube, Y. Kajikawa, and S. Miyoshi, "Statistical-mechanical analysis of the second-order adaptive volterra filter," Asia-Pacific Signal and Information Processing Association Annual Summit and Conf. (APSIPA ASC), pp.1821–1824, 2018.

Kimiko Motonaka received her B.E. degree from the Department of System Engineering in 2010, and M.E. and D.E. degrees from the Department of Natural Science and Technology in 2012 and 2014 of Okayama University, respectively. She is currently an assistant professor at the Department of Electrical and Electronic Engineering of Kansai University. Her research interests include motion planning and group control for robots. Dr. Motonaka is a member of the IEEE.

Tomoya Koseki received his B.E. degree in Electrical, Electronic and Information Engineering from the Faculty of Engineering Science, Kansai University in 2018. He is currently working at JB Toll Systems Corporation.

Yoshinobu Kajikawa received his B.Eng. and M.Eng. degrees in electrical engineering from Kansai University, Osaka, Japan, in 1991 and 1993, respectively. He received his D.Eng. degree in communication engineering from Osaka University, Osaka, Japan, in 1997. He joined Fujitsu Ltd., Kawasaki, Japan, in 1993, and engaged in research on active noise control. In 1994, he joined Kansai University, Osaka, Japan, where he is currently a professor. His current research interests include signal processing for audio and acoustic systems. He has authored or coauthored more than 200 articles in journals and conference proceedings, and has more than 10 patents. He received the 2012 Sato Prize Paper Award from the Acoustical Society of Japan (ASJ), the Best Paper Award in APCCAS 2014, the 2017 Sadaoki Furui Prize Paper Award from the Asia Pacific Signal and Information Processing Association (APSIPA), and the 2019 Best Paper Award from IEICE. He is a senior member of IEEE and IEICE, and a member of APSIPA, Acoustical Society of America (ASA), and ASJ. He is now serving as the Editor-in-Chief of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. He is also an associate editor of the IET Signal Processing and Applied Sciences, and a member of SLA TC in APSIPA. He is now a vice president of APSIPA and a vice chair of TC on Engineering Acoustics in IEICE. He was a vice president of IEICE Engineering and Science Society.

Seiji Miyoshi received his B.Eng. and M.Eng. degrees in electrical engineering from Kyoto University, Japan, in 1986 and 1988, respectively, and his Ph.D degree in system science and engineering from Kanazawa University, Japan, in 1998. He worked with the Space Development Division of NEC Corporation from 1988 to 1994. He joined the Department of Electronic Engineering of Kobe City College of Technology in 1994. He joined the Department of Electrical and Electronic Engineering, the Faculty of Engineering Science of Kansai University in 2008, where he is now a professor. His research interests include statistical-mechanical informatics, image processing, signal processing, and learning theory. Dr. Miyoshi is a senior member of the Institute of Electronics, Information and Communication Engineers (IEICE) and the Institute of Electrical and Electronics Engineers (IEEE), and a member of the Physical Society of Japan (JPS), and the Japan Neural Network Society (JNNS). He served as an associate editor for IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences from 2013 to 2017 and an awards committee member of IEEE Kansai section from 2011 to 2014. He received the 2011 Papers of Editors' Choice from the Journal of the Physical Society of Japan, the Excellent Paper Award in ICONIP2016, and the 2019 Best Paper Award from IEICE.