

Bayesian Estimation of Multi-Trap RTN Parameters Using Markov Chain Monte Carlo Method

Hiromitsu AWANO^{†a)}, Nonmember, Hiroshi TSUTSUI^{†b)}, Hiroyuki OCHI^{†c)}, and Takashi SATO^{†d)}, Members

SUMMARY Random telegraph noise (RTN) is a phenomenon that is considered to limit the reliability and performance of circuits using advanced devices. The time constants of carrier capture and emission and the associated change in the threshold voltage are important parameters commonly included in various models, but their extraction from time-domain observations has been a difficult task. In this study, we propose a statistical method for simultaneously estimating interrelated parameters: the time constants and magnitude of the threshold voltage shift. Our method is based on a graphical network representation, and the parameters are estimated using the Markov chain Monte Carlo method. Experimental application of the proposed method to synthetic and measured time-domain RTN signals was successful. The proposed method can handle interrelated parameters of multiple traps and thereby contributes to the construction of more accurate RTN models.

key words: random telegraph noise, Bayesian estimation, Markov chain Monte Carlo, device characterization, source separation, statistical machine learning

1. Introduction

Random telegraph noise (RTN) is the temporal change in threshold voltages observed in metal-oxide-semiconductor (MOS) transistors. Figure 1 shows the RTN generation mechanism. The capture of a carrier to an electrically active trap (defect) and its release cause a charge change in the dielectric, leading to shifts in the threshold voltage [1]. As device sizes shrink, the effects of RTN are expected to become more serious [2].

Nagumo et al. introduced a time lag plot (TLP) to investigate the magnitude of the threshold voltage shift caused by RTN [3]. They further analyzed other parameters such as trap position by measuring devices with a single observable trap [4]. Miki et al. [5] and Realov et al. [6] applied a hidden Markov model (HMM) for investigating the time constants of RTN. In the conventional studies, however, extraction of amplitudes has been limited to devices that have equal to or less than two traps. Since devices having more than two traps in measurement data are common, it is important to develop a method that is applicable to arbitrary number of traps. Furthermore, the extraction of time constants cannot be conducted through HMM. Hence, a sep-

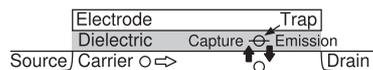


Fig. 1 Generation process of RTN.

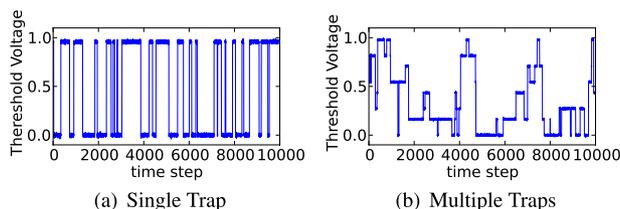


Fig. 2 Two examples of threshold voltage fluctuation caused by RTN.

arate process, such as application of TLP, is required. Application of two different analyses to extract interrelated parameters leads to insufficient accuracy in parameter extraction. In Table 1, characteristics of the proposed and the existing methods are compared. Figure 2 shows two examples of measured threshold voltage shifts due to RTN. In the case of devices with a single trap, we can estimate the magnitude of the threshold voltage shift simply by calculating the distance between two peaks in threshold voltage histogram. On the other hand, in the case of multiple traps, parameter extraction becomes much more complicated because parameters such as magnitudes, time constants, and trap states are strongly interrelated. Furthermore, it is an under-determined problem because it requires reconstruction of multi-trap time series sequences from only the threshold voltage fluctuation, which is superimposed with multi-trap activity. Hence, the direct use of HMM, as in conventional estimation methods, is inappropriate.

In this paper, we propose a novel method for extracting RTN parameters based on a machine learning method. We construct a statistical generation model that reflects the device physics of RTN, in which magnitudes, time constants, and trap states are random parameter variables. In our method, these parameters are estimated using Markov chain Monte Carlo methods (MCMC) so that they best fit the model to an observed RTN signal.

The proposed method has advantages over existing methods, which are summarized as follows.

1. There is no limit to the number of traps. More than two traps can be treated by the proposed method.
2. All RTN parameters are directly estimated. No post-processing is necessary.

Manuscript received March 19, 2012.

Manuscript revised June 18, 2012.

[†]The authors are with the Department of Communications and Computer Engineering, the Graduate School of Informatics, Kyoto University, Kyoto-shi, 606-8501 Japan.

a) E-mail: awano@easter.kuee.kyoto-u.ac.jp

b) E-mail: tsutsui@easter.kuee.kyoto-u.ac.jp

c) E-mail: ochi@kuee.kyoto-u.ac.jp

d) E-mail: takashi@i.kyoto-u.ac.jp

DOI: 10.1587/transfun.E95.A.2272

Table 1 Summary of capabilities among the proposed and the existing methods.

	Number of traps	Amplitudes extraction	Time constants extraction
Proposed Method	Can be applied to more than two traps	OK	OK
TLP [3]	Equal to or less than two traps	OK	NG
TLP+HMM [5], [6]	Equal to or less than two traps	OK	OK (insufficient accuracy)

- The RTN parameters are simultaneously estimated so that estimations of the interrelated parameters become consistent.

The rest of this paper is organized as follows. In Sect. 2, we will describe the proposed method for decomposing the statistics of traps. In Sect. 3, we will describe the experimental validation of the method using synthetic RTN and its results. We will also discuss the ability of the method. In Sect. 4, we will describe the results of a parameter extraction experiment using measured threshold voltages. Finally, Sect. 6 summarizes this paper.

2. Proposed Method

In this section, we describe the proposed model and parameter extraction method.

2.1 Problem Setting

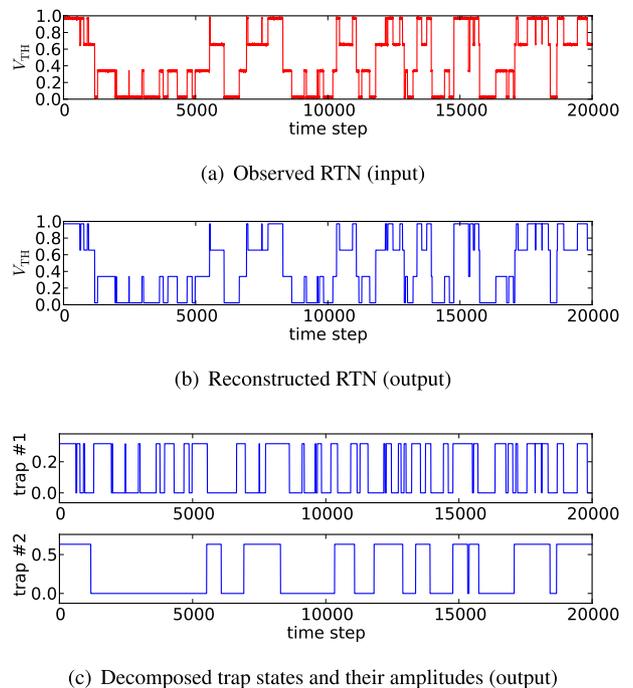
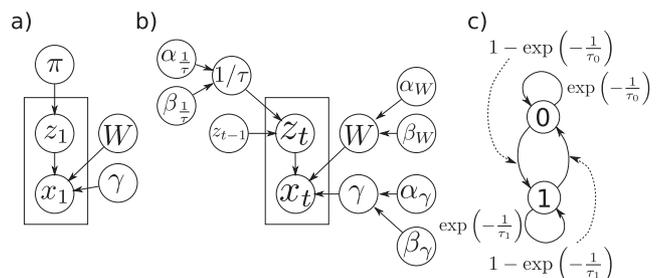
The input and output of the proposed method is summarized as follows.

- Input: measured RTN and the maximum number of traps assumed.
- Output: estimated temporal sequence of trap states, magnitudes of threshold voltage shift, and time constants of the traps.

The inputs of the proposed method are the threshold voltages as a function of time and the maximum number of traps assumed. The determination of the number of traps will also be described in Sect. 2.3. Figure 3(a) shows an example of the measured threshold voltage time series. The outputs of the proposed method are the RTN parameters, i.e., the estimated trap states and magnitude of the threshold voltage shifts caused by the respective traps, as shown in Fig. 3(c). Figure 3(b) shows the reconstructed threshold voltage shift using the estimated magnitudes and trap states.

2.2 Proposed Statistical Generation Model

Figure 4 shows the graphical model defined in the proposed method. Each node corresponds to a random variable. The links among nodes represent the relationships among the random variables. The variable w which represent magnitudes of threshold voltage shifts, for example, depend on α_w and β_w . The trap states represented by the binary latent variables $z_{(t,i)}$, where $t = 1, \dots, N$ are the time steps, $i = 1, \dots, K$ are the indices of the traps, N is the number of observations, and K is the number of traps. The threshold voltages at each time step are $X = \{x_1, x_2, \dots, x_N\}$.

**Fig. 3** Example input and outputs of the proposed method.**Fig. 4** Proposed graphical model representing the generation process of RTN. a) $t = 1$, b) transition of a state from $t - 1$ to t , and c) modeling of a trap.

The total threshold voltage shift is modeled as a linear summation of the threshold voltage shifts caused by each trap, which in turn is modeled as the product of the trap state and corresponding magnitude. Here, we assume that the observation error follows a normal distribution whose standard deviation is $\sqrt{\frac{1}{\gamma}}$. The probability distribution of x_t can be written as

$$p(x_t | z_{(t,1:K)}, w_{1:K}, \gamma) = \mathcal{N}\left(x_t \mid \sum_{i=1}^K w_i z_{(t,i)}, \sqrt{\frac{1}{\gamma}}\right), \quad (1)$$

where $w_{1:K}$ is the magnitude of the trap and $\mathcal{N}(x|\mu, \sigma)$ represents a probability distribution function of the normal dis-

tribution defined as

$$\mathcal{N}(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right). \quad (2)$$

The probability distribution of z_t depends on z_{t-1} , which is the state of the previous time. The conditional distribution $p(z_{(t,i)}|z_{(t-1,i)}, \frac{1}{\tau})$ can be written as

$$p(z_{(t,i)}|z_{(t-1,i)}) = \begin{cases} \exp(-1/\tau_{(i,0)}) & z_{(t-1,i)} = 0, z_{(t,i)} = 0 \\ 1 - \exp(-1/\tau_{(i,0)}) & z_{(t-1,i)} = 0, z_{(t,i)} = 1 \\ 1 - \exp(-1/\tau_{(i,1)}) & z_{(t-1,i)} = 1, z_{(t,i)} = 0 \\ \exp(-1/\tau_{(i,1)}) & z_{(t-1,i)} = 1, z_{(t,i)} = 1 \end{cases}. \quad (3)$$

The initial latent node z_1 is special because it does not depend on the previous node, so we give z_1 a marginal distribution $p(z_{(1,i)}|\pi_i)$ defined as

$$p(z_{(1,i)}|\pi_i) = \pi_i^{1-z_{(1,i)}} (1-\pi_i)^{z_{(1,i)}}. \quad (4)$$

Each model parameter has the following prior distribution

$$p(w_i|\theta_{w_i}) = \mathcal{G}(w_i|\alpha_{w_i}, \beta_{w_i}), \quad (5)$$

$$p(1/\tau_i|\theta_{1/\tau_i}) = \mathcal{G}(1/\tau_i|\alpha_{1/\tau_i}, \beta_{1/\tau_i}), \quad (6)$$

$$p(\gamma|\theta_\gamma) = \mathcal{G}(\gamma|\alpha_\gamma, \beta_\gamma), \quad (7)$$

$$p(\pi_i|\theta_\pi) = \mathcal{B}(\pi_i|\alpha_\pi, \beta_\pi), \quad (8)$$

$$p(\alpha_{w_i}|\lambda_{\alpha_{w_i}}) = \mathcal{L}(\alpha_{w_i}|\lambda_{\alpha_{w_i}}), \quad (9)$$

$$p(\beta_{w_i}|\lambda_{\beta_{w_i}}) = \mathcal{L}(\beta_{w_i}|\lambda_{\beta_{w_i}}), \quad (10)$$

$$p(\alpha_{1/\tau_i}|\lambda_{\alpha_{1/\tau_i}}) = \mathcal{L}(\alpha_{1/\tau_i}|\lambda_{\alpha_{1/\tau_i}}), \text{ and} \quad (11)$$

$$p(\beta_{1/\tau_i}|\lambda_{\beta_{1/\tau_i}}) = \mathcal{L}(\beta_{1/\tau_i}|\lambda_{\beta_{1/\tau_i}}), \quad (12)$$

where $\mathcal{G}(x|\alpha, \beta)$, $\mathcal{B}(x|\alpha, \beta)$, and $\mathcal{L}(x|\lambda)$ are the gamma, beta, and exponential distributions, respectively. They are defined as

$$\mathcal{G}(x|\alpha, \beta) = \frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} \exp\left(-\frac{x}{\beta}\right) \quad x > 0, \quad (13)$$

$$\mathcal{B}(x|\alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}, \text{ and} \quad (14)$$

$$\mathcal{L}(x|\lambda) = \lambda \exp(-\lambda x) \quad x \geq 0. \quad (15)$$

Here, $B(\alpha, \beta)$ and $\Gamma(\alpha)$ are the beta function and the gamma function defined as

$$B(\alpha, \beta) = \int_0^1 t^{\alpha-1} (1-t)^{\beta-1} dt \text{ and} \\ \Gamma(\alpha) = \int_0^\infty e^{-t} t^{\alpha-1} dt, \text{ respectively.} \quad (16)$$

2.3 Determination of the Number of Traps

The proposed method automatically adjusts the model complexity based on Bayesian inference. Owing to the sparse assumption placed to the prior distribution, excess traps

will be degenerated, i.e., amplitudes of those traps that are unnecessary to explain a given RTN waveform converge around zero. By comparing the estimated magnitude of noise and the amplitude of a trap, we can determine the most appropriate number of traps. Hence, when we apply the proposed method to an RTN waveform having unknown number of traps, we may give a large number, such as five, as the initial number of traps. Alternatively, we can also loosely estimate the number of traps using TLP.

2.4 Parameter Estimation Algorithm

To estimate the model parameters, we have to evaluate the posterior distribution

$$P(w, 1/\tau, \gamma, Z, \alpha_w, \beta_w, \alpha_{1/\tau}, \beta_{1/\tau} | X), \quad (17)$$

which represents the model parameter probability after X is observed. However, the posterior distribution cannot be analytically calculated because this requires analytical integration of the posterior density function. Hence, we approximate the posterior distribution by samples generated from the distribution using the MCMC method.

2.4.1 Gibbs Sampling

Gibbs sampling is an example of an MCMC algorithm. The purpose of this algorithm is to generate a sequence of samples from a joint probability distribution of multivariate random variables [7]. Each step in the Gibbs sampling algorithm involves replacing the value of one random variable with a new sample generated from the distribution conditioned on the values of the remaining variables. Here, we outline how Gibbs sampling can be applied to a Monte Carlo approximation of Eq. (17). The following procedures are iterated after initialization of each random variables, and $w^{(n)}$ indicates the samples drawn from the corresponding distribution in the n -th iteration of Gibbs sampling.

1. For $i = 1, \dots, K$, sample w_i from the following conditional posterior density,

$$w_i^{(n+1)} \sim p\left(w_i \mid x_{(1:N)}, w_{(1:i-1)}^{(n+1)}, w_{(i+1:K)}^{(n)}, z_{(1:N)}^{(n)}, \gamma^{(n)}, \alpha_{w_i}^{(n)}, \beta_{w_i}^{(n)}\right). \quad (18)$$

2. Sample $\gamma^{(n+1)}$ from the following conditional posterior density,

$$\gamma^{(n+1)} \sim p\left(\gamma \mid x_{(1:N)}, w_{(1:K)}^{(n+1)}, z_{(1:N, 1:K)}^{(n)}, \alpha_\gamma^{\text{prior}}, \beta_\gamma^{\text{prior}}\right). \quad (19)$$

Note that the new sample of w_i drawn at the previous step is used.

3. Sample $z_1^{(n+1)}$ from the following posterior density,

$$z_1^{(n+1)} \sim p\left(z_1 \mid z_2^{(n)}, x_1, w_{(1:N)}^{(n+1)}, \gamma^{(n+1)}, 1/\tau^{(n)}, \pi^{(n)}\right). \quad (20)$$

In this step, w and γ are replaced with the new samples

$w^{(n+1)}$ and $\gamma^{(n+1)}$.

4. For $i = 1, \dots, K$, sample $\pi_i^{(n+1)}$ from the following posterior density,

$$\pi_i^{(n+1)} \sim p\left(\pi_i \mid z_{(1,k)}^{(n+1)}, \alpha_{\pi_i}^{\text{prior}}, \beta_{\pi_i}^{\text{prior}}\right). \quad (21)$$

5. For $t = 2, \dots, N$, sample $z_t^{(n+1)}$ from the following posterior density,

$$z_t^{(n+1)} \sim p\left(z_t \mid x_t, z_{t-1}^{(n+1)}, z_{t+1}^{(n)}, 1/\tau^{(n)}, w_{(1:K)}^{(n+1)}, \gamma^{(n+1)}\right). \quad (22)$$

6. For $i = 1, \dots, K$ and $l = 0, 1$, sample $\frac{1}{\tau_{(i,l)}}$ from the following posterior density,

$$1/\tau_{(i,l)}^{(n+1)} \sim p\left(1/\tau_{(i,l)} \mid z_{(1:N,i)}^{(n+1)}, \alpha_{1/\tau_{(i,l)}}^{(n)}, \beta_{1/\tau_{(i,l)}}^{(n)}\right). \quad (23)$$

7. For $i = 1, \dots, K$, sample $\alpha_{w_i}^{(n+1)}$ and $\beta_{w_i}^{(n+1)}$ from the following posterior densities, respectively,

$$\alpha_{w_i}^{(n+1)} \sim p\left(\alpha_{w_i} \mid w_i^{(n+1)}, \beta_{w_i}^{(n)}, \lambda_{\alpha_{w_i}}\right) \text{ and} \quad (24)$$

$$\beta_{w_i}^{(n+1)} \sim p\left(\beta_{w_i} \mid w_i^{(n+1)}, \alpha_{w_i}^{(n+1)}, \lambda_{\beta_{w_i}}\right). \quad (25)$$

8. For $i = 1, \dots, K$ and $l = 0, 1$, sample $\alpha_{1/\tau_{(i,l)}}$ and $\beta_{1/\tau_{(i,l)}}$ in the same way as sampling α_{w_i} and β_{w_i} .

2.4.2 Conditional Posterior Density

A part of developing the posterior density is derived from a paper on sound source separation using MCMC [8].

1. The posterior density corresponding to $w_i^{(n+1)}$ is

$$\begin{aligned} p\left(w_i \mid X, w_{(1:i-1)}^{(n+1)}, w_{(i+1:K)}^{(n)}, z_{(1:N)}^{(n)}, \gamma^{(n)}, \alpha_{w_i}^{(n)}, \beta_{w_i}^{(n)}\right) \\ \propto p\left(x_{(1:N)} \mid z_{(1:N,1:K)}^{(n)}, w_{(1:i-1)}^{(n+1)}, w_i, w_{(i+1:K)}^{(n)}, \gamma^{(n)}\right) \\ \times p\left(w_i \mid \alpha_{w_i}, \beta_{w_i}\right). \end{aligned} \quad (26)$$

The first term on the right-hand side of Eq. (26) can be written as

$$\begin{aligned} p\left(x_{(1:N)} \mid z_{(1:N,1:K)}^{(n)}, w_{(1:i-1)}^{(n+1)}, w_i, w_{(i+1:K)}^{(n)}, \gamma^{(n)}\right) \\ \propto \exp\left\{-\frac{\gamma_{w_i}^{\text{likel}}}{2} \left(w_i - \mu_{w_i}^{\text{likel}}\right)^2\right\}, \end{aligned} \quad (27)$$

where $\gamma_{w_i}^{\text{likel}}$, $\mu_{w_i}^{\text{likel}}$, and $\mathcal{E}_{(t)}^{(-i)}$ are defined as

$$\gamma_{w_i}^{\text{likel}} = \gamma^{(n)} \left(\sum_{t=1}^N z_{(t,i)}^{(n)2} \right), \quad (28)$$

$$\mu_{w_i}^{\text{likel}} = \frac{\sum_{t=1}^N z_{(t,i)}^{(n)} \mathcal{E}_{(t)}^{(-i)}}{\sum_{t=1}^N z_{(t,i)}^{(n)2}}, \text{ and} \quad (29)$$

$$\mathcal{E}_{(t)}^{(-i)} = x_t - \sum_{j=1}^{i-1} z_{(t,j)}^{(n)} w_j^{(n+1)} - \sum_{j=i+1}^K z_{(t,j)}^{(n)} w_j^{(n)}. \quad (30)$$

The second term on the right-hand side of Eq. (26) is the probability density function of the gamma distribution defined in Eq. (13). Therefore, the posterior density function can be written as

$$\begin{aligned} p\left(w_i \mid X, w_{(1:i-1)}^{(n+1)}, w_{(i+1:K)}^{(n)}, z_{(1:N)}^{(n)}, \gamma^{(n)}, \alpha_{w_i}^{(n)}, \beta_{w_i}^{(n)}\right) \\ \propto w_i^{\alpha_{w_i}^{(n)}-1} \exp\left\{-\frac{\gamma_{w_i}^{\text{likel}}}{2} \left(w_i - \mu_{w_i}^{\text{likel}}\right)^2 - \frac{w_i}{\beta_{w_i}^{(n)}}\right\}. \end{aligned} \quad (31)$$

The shape of this distribution is too complex to analytically calculate the normalization constant, which requires analytical integration of the distribution. Hence, we sample w_i by using the Metropolis method [9]. To construct a proposal distribution, we approximate the posterior distribution by a Gaussian distribution whose variance and mode is same as the posterior distribution. First, Eq. (31) is rewritten as follows,

$$\begin{aligned} p\left(w_i \mid X, w_{(1:i-1)}^{(n+1)}, w_{(i+1:K)}^{(n)}, z_{(1:N)}^{(n)}, \gamma^{(n)}, \alpha_{w_i}^{(n)}, \beta_{w_i}^{(n)}\right) \\ \propto w_i^{\alpha_{w_i}^{(n)}-1} \exp\left\{-\frac{\left(w_i - \mu_{w_i}^{\text{post}}\right)^2 \gamma_{w_i}^{\text{post}}}{2}\right\}, \end{aligned} \quad (32)$$

where $\gamma_{w_i}^{\text{post}}$ and $\mu_{w_i}^{\text{post}}$ are defined as

$$\gamma_{w_i}^{\text{post}} = \gamma_{w_i}^{\text{likel}} \text{ and} \quad (33)$$

$$\mu_{w_i}^{\text{post}} = \mu_{w_i}^{\text{likel}} - \frac{1}{\beta_{w_i}^{(n)} \gamma_{w_i}^{\text{likel}}}. \quad (34)$$

The mode of the posterior density can be calculated as the solution of the following equation,

$$\begin{aligned} w_i^{\alpha_{w_i}^{(n)}-2} \exp\left\{-\frac{\left(w_i - \mu_{w_i}^{\text{post}}\right)^2 \gamma_{w_i}^{\text{post}}}{2}\right\} \\ \times \left(w_i^2 - \mu_{w_i}^{\text{post}} w_i - \frac{\alpha_{w_i}^{(n)} - 1}{\gamma_{w_i}^{\text{post}}}\right) = 0. \end{aligned} \quad (35)$$

Therefore, the mode of the posterior distribution can be written as

$$\mu_{w_i}^{\text{max}} = \begin{cases} 0 & D < 0 \\ \max\left\{\frac{1}{2} \left(\mu_{w_i}^{\text{post}} + \sqrt{D}\right), 0\right\} & \text{otherwise} \end{cases} \quad (36)$$

$$D = \left(\mu_{w_i}^{\text{post}}\right)^2 + 4 \frac{\alpha_{w_i}^{(n)} - 1}{\gamma_{w_i}^{\text{post}}}. \quad (37)$$

Finally, we can obtain the following proposal distribution,

$$q(w_i) = \mathcal{N}\left(w_i^{(n+1)} \mid \mu_{w_i}^{\text{max}}, \sqrt{1/\gamma_{w_i}^{\text{post}}}\right). \quad (38)$$

2. The posterior density corresponding to $\gamma^{(n+1)}$ is

$$\begin{aligned} p\left(\gamma \mid x_{(1:N)}, z_{(1:N,1:K)}^{(n)}, w_{(1:K)}^{(n+1)}, \alpha_{\gamma}^{\text{prior}}, \beta_{\gamma}^{\text{prior}}\right) \\ \propto p\left(x_{(1:N)} \mid z_{(1:N,1:K)}^{(n)}, w_{(1:K)}^{(n+1)}, \gamma\right) \end{aligned}$$

$$\times p\left(\gamma | \alpha_\gamma^{\text{prior}}, \beta_\gamma^{\text{prior}}\right). \quad (39)$$

The first term on the right-hand side of Eq. (39) can be written as

$$\begin{aligned} & p\left(x_{(1:N)} | z_{(1:N,1:K)}^{(n)}, w_{(1:K)}^{(n+1)}, \gamma\right) \\ &= \prod_{t=1}^N \mathcal{N}\left(x_t \mid \sum_{i=1}^K z_{(t,i)}^{(n)} w_i^{(n+1)}, \gamma\right) \\ &\propto \gamma^{N/2} \exp\left\{-\frac{\gamma}{2} \sum_{t=1}^N \left(x_t - \sum_{i=1}^K z_{(t,i)}^{(n)} w_i^{(n+1)}\right)^2\right\}. \quad (40) \end{aligned}$$

Because the prior distribution $p\left(\gamma | \alpha_\gamma^{\text{prior}}, \beta_\gamma^{\text{prior}}\right)$ is a gamma distribution, the posterior distribution can also be written as the following gamma distribution,

$$\begin{aligned} & p\left(\gamma^{(n+1)} | x_{(1:N)}, z_{(1:N,1:K)}^{(n)}, w_{(1:K)}^{(n+1)}, \alpha_\gamma^{\text{prior}}, \beta_\gamma^{\text{prior}}\right) \\ &= \mathcal{G}\left(\gamma^{(n+1)} | \alpha_\gamma^{\text{post}}, \beta_\gamma^{\text{post}}\right), \quad (41) \end{aligned}$$

where

$$\alpha_\gamma^{\text{post}} = \alpha_\gamma^{\text{prior}} + \frac{N}{2} \quad \text{and} \quad (42)$$

$$\beta_\gamma^{\text{post}} = \left\{ \frac{1}{\beta_\gamma^{\text{prior}}} + \frac{1}{2} \sum_{t=1}^N \left(x_t - \sum_{j=1}^K z_{(t,j)}^{(n)} w_j^{(n)}\right)^2 \right\}^{-1}. \quad (43)$$

3. The posterior distribution corresponding to $z_t^{(n+1)}$ is

$$\begin{aligned} & p\left(z_t | x_t, z_{t-1}^{(n+1)}, z_{t+1}^{(n)}, 1/\tau^{(n)}, w_{(1:K)}^{(n+1)}, \gamma^{(n+1)}\right) \\ &\propto p\left(x_t | z_t, w_{(1:K)}^{(n+1)}, \gamma^{(n+1)}\right) p\left(z_t | z_{t-1}^{(n+1)}, 1/\tau^{(n)}\right) \\ &\quad \times p\left(z_{t+1}^{(n)} | z_t, 1/\tau^{(n)}\right), \quad (44) \end{aligned}$$

where $p\left(x_t | z_t, w_{(1:K)}, \gamma\right)$ and $p\left(z_{(t,i)} | z_{(t-1,i)}, 1/\tau\right)$ are defined in Eq. (1) and Eq. (3), respectively. To sample z_t , we enumerate the possible combinations of z_t (number of combinations: 2^K), calculate the probability of each combination, and sample z_t according to the probability.

4. The posterior distribution corresponding to $z_1^{(n+1)}$ is

$$\begin{aligned} & p\left(z_1 | x_1, z_2^{(n)}, 1/\tau^{(n)}, w_{(1:K)}^{(n+1)}, \gamma^{(n+1)}\right) \\ &\propto p\left(x_1 | z_1, w_{(1:K)}^{(n+1)}, \gamma^{(n+1)}\right) \\ &\quad \times p\left(z_2^{(n)} | z_1, 1/\tau^{(n)}\right) p\left(z_1 | \pi^{(n)}\right), \quad (45) \end{aligned}$$

where $p\left(z_1 | \pi^{(n)}\right)$ is defined as Eq. (4). Drawing samples from the posterior distribution is done in the same way as sampling z_t ($n+1$).

5. The posterior distribution corresponding to $1/\tau_{(i,l)}^{(n+1)}$ is

$$p\left(\frac{1}{\tau_{(i,l)}} \mid z_{1:N,i}^{(n+1)}, \alpha_{1/\tau_{(i,l)}}^{(n)}, \beta_{1/\tau_{(i,l)}}^{(n)}\right)$$

$$\propto p\left(z_{(1:N,i)}^{(n+1)} \mid \frac{1}{\tau_{(i,l)}}\right) p\left(\frac{1}{\tau_{(i,l)}} \mid \alpha_{1/\tau_{(i,l)}}^{(n)}, \beta_{1/\tau_{(i,l)}}^{(n)}\right). \quad (46)$$

The first term on the right-hand side of Eq. (46) can be written as

$$\begin{aligned} & p\left(z_{(1:N,i)}^{(n+1)} \mid \frac{1}{\tau_{(i,l)}}\right) \\ &= \exp\left(-\frac{1}{\tau_{(i,l)}}\right)^{n_{l \rightarrow l}} \left\{1 - \exp\left(-\frac{1}{\tau_{(i,l)}}\right)\right\}^{n_{l \rightarrow \bar{l}}} \quad (47) \end{aligned}$$

where $n_{l \rightarrow l}$ is the number of steps whose state is the same as the previous one ($z_{(t,i)}^{(n+1)} = z_{(t+1,i)}^{(n+1)} = l$), and $n_{l \rightarrow \bar{l}}$ is the number of steps whose state is different from the previous one ($z_{(t,i)}^{(n+1)} = l$ and $z_{(t+1,i)}^{(n+1)} \neq l$). The second term on the right-hand side of Eq. (46) is the gamma distribution. The posterior distribution does not belong to well-known distributions. Therefore, samples are drawn using the Metropolis method.

6. The posterior distribution corresponding to $\pi_i^{(n+1)}$ is

$$\begin{aligned} & p\left(\pi_i | z_{(1,i)}^{(n+1)}, \alpha_{\pi_i}^{\text{prior}}, \beta_{\pi_i}^{\text{prior}}\right) \\ &\propto p\left(z_{(1,i)}^{(n+1)} | \pi_i\right) p\left(\pi_i | \alpha_{\pi_i}^{\text{prior}}, \beta_{\pi_i}^{\text{prior}}\right). \quad (48) \end{aligned}$$

The first term on the right-hand side of Eq. (48) is defined in Eq. (4). The second term on the right-hand side of Eq. (48) is the beta distribution defined in Eq. (15). Hence, the posterior distribution also becomes the following beta distribution,

$$p\left(\pi_i | z_{(1,i)}^{(n+1)}, \alpha_{\pi_i}^{\text{prior}}, \beta_{\pi_i}^{\text{prior}}\right) = \mathcal{B}\left(\pi_i | \alpha_{\pi_i}^{\text{post}}, \beta_{\pi_i}^{\text{post}}\right), \quad (49)$$

where

$$\alpha_{\pi_i}^{\text{post}} = \alpha_{\pi_i}^{\text{prior}} - z_{(1,i)}^{(n+1)} + 1 \quad \text{and} \quad (50)$$

$$\beta_{\pi_i}^{\text{post}} = \beta_{\pi_i}^{\text{prior}} + z_{(1,i)}^{(n+1)}. \quad (51)$$

7. The posterior distribution corresponding to $\alpha_{w_i}^{(n+1)}$ is

$$\begin{aligned} & p\left(\alpha_{w_i} | w_i^{(n+1)}, \beta_{w_i}^{(n)}, \lambda_{\alpha_{w_i}}\right) \\ &\propto p\left(w_i^{(n+1)} | \alpha_{w_i}, \beta_{w_i}^{(n)}\right) p\left(\alpha_{w_i} | \lambda_{\alpha_{w_i}}\right) \\ &= \mathcal{G}\left(w_i^{(n+1)} | \alpha_{w_i}, \beta_{w_i}^{(n)}\right) \mathcal{L}\left(\alpha_{w_i} | \lambda_{\alpha_{w_i}}\right). \quad (52) \end{aligned}$$

This posterior distribution also does not belong to well-known distributions. Therefore, the Metropolis method is used to draw samples from the posterior distribution.

8. The posterior distribution for sampling β_{w_i} , $\alpha_{1/\tau_{(i,l)}}$ and $\beta_{1/\tau_{(i,l)}}$ is the same as the posterior distribution of α_{w_i} .

2.4.3 Proposed Parameter Estimation Algorithm

The proposed parameter estimation algorithm is defined as below.

1. Initialize the following random variables: $z_{(1:N,1:K)}^{(0)}$, $\gamma^{(0)}$, $w_{(1:K)}^{(0)}$, $1/\tau_{(1:K,0;1)}^{(0)}$, $\pi_{1:K}^{(0)}$, $\alpha_{w_{1:K}}^{(0)}$, $\beta_{w_{1:K}}^{(0)}$, $\alpha_{1/\tau_{(1:K,0;1)}}^{(0)}$, and

$$\beta_{1/\tau_{(1:K,0:1)}}^{(0)}.$$

2. Repeat the following steps until generated samples become independent of the initial values.

- a. Sample $w_i^{(n+1)}$ for $i = 1, \dots, K$:
draw a candidate sample of $w_i^{(n+1)}$ from Eq. (38) (w^*). Sample u from a uniform distribution. Accept w^* as $w_i^{(n+1)}$ if $\min\left(1, \frac{p(w^*)}{p(w_i^{(n)})}\right) > u$. Otherwise, the candidate sample is rejected and $w_i^{(n+1)}$ is set to $w_i^{(n)}$.
- b. Sample $\gamma^{(n+1)}$:
draw a sample from Eq. (41) conditioned on $X, z_{(1:N,1:K)}^{(n)}, w_{(1:K)}^{(n+1)}, \alpha_\gamma$, and β_γ .
- c. Sample $z_1^{(n+1)}$:
calculate the probabilities of each possible combination of z_1 by Eq. (45). Calculate the cumulative probability and sample $z_1^{(n+1)}$ according to a random value drawn from uniform distribution.
- d. Sample π_i for $i = 1, \dots, K$:
draw a sample from Eq. (49) conditioned on $z_1^{(n+1)}, \alpha_{\pi_i}$, and β_{π_i} .
- e. Sample $z_t^{(n+1)}$ for $t = 2, \dots, N$:
calculate the cumulative probability by Eq. (44) and sample z_t in the same manner as sampling z_1 .
- f. Sample $\frac{1}{\tau_{(i,l)}^{(n+1)}}$ for $i = 1, \dots, K, l = 0, 1$:
draw a sample from Eq. (46) conditioned on $z_{(1:N,1:K)}^{(n+1)}, \alpha_{1/\tau_{(i,l)}^{(n)}},$ and $\beta_{1/\tau_{(i,l)}^{(n)}}$.
- g. Sample $\alpha_{w_i}^{(n+1)}$ for $i = 1, \dots, K$:
draw a sample from Eq. (52) conditioned on $w_i^{(n+1)}, \beta_{w_i}^{(n)},$ and $\lambda_{\alpha_{w_i}}$.
- h. Sample $\beta_{w_i}^{(n+1)}$ for $i = 1, \dots, K$:
draw a sample from Eq. (52) conditioned on $w_i^{(n+1)}, \alpha_{w_i}^{(n+1)},$ and $\lambda_{\beta_{w_i}}$.
- i. Sample $\alpha_{1/\tau_{(i,l)}}^{(n+1)}$ and $\beta_{1/\tau_{(i,l)}}^{(n+1)}$:
draw samples in the same way as sampling α_{w_i} and β_{w_i} , respectively.

3. Experimental Validation

3.1 Preliminary Experiment Using Synthetic RTN Data

To validate the proposed estimation algorithm, we first conduct an experiment with a synthetic RTN signal. We generate a time series signal with 100,000 steps that simulates the threshold voltage shift caused by RTN. Three traps are used for the simulated RTN data, and the amplitudes and time constants are set as listed in Table 2. The parameters of the prior distributions are set to $\alpha_\gamma = 10^6, \beta_\gamma = 10^{-3}, \lambda_{\alpha_w} = 0.1, \lambda_{\beta_w} = 0.1, \lambda_{\alpha_{1/\tau}} = 1.0, \lambda_{\beta_{1/\tau}} = 1.0, \alpha_\pi = 1.0,$ and $\beta_\pi = 1.0$. The estimated parameters are listed in Table 3. Figure 5 shows the estimated trap states and reconstructed RTN using these estimated parameters. The first 20,000 steps of the time series are plotted.

Comparing Table 3 with Table 2, we can see that the

Table 2 Parameters used for generating synthetic RTN data.

Trap	#1	#2	#3
Amplitude	0.603	0.957	1.84
τ_0	300	500	800
τ_1	300	500	800

Table 3 Estimated parameters using the proposed method from the synthetic RTN data. Note that trap numbers are arbitrary compared with Table 2.

Trap	#1	#2	#3
Amplitude	1.72	0.862	0.518
τ_0	712	571	306
τ_1	696	473	288

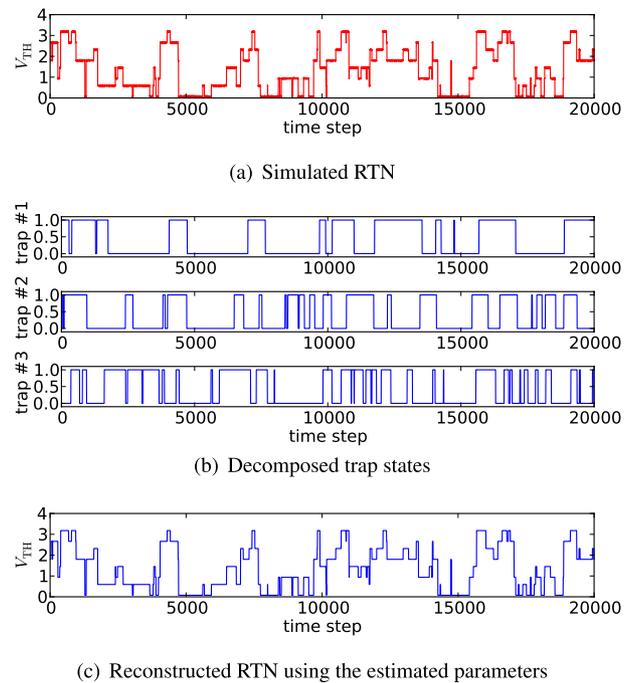


Fig. 5 Estimated trap states and reconstructed RTN signal. The time change of the three trap states and their amplitudes are cleanly extracted in (b). Good agreement is observed between the estimation in (c) and the input data in (a).

proposed method successfully estimates the magnitude and time constants of the traps of the synthetic RTN signal within an error of 13.9% and 14.2%, respectively. From Fig. 5, we can see that the amplitude and trap states are extracted with good accuracy from the input data. They match the results obtained from a manual extraction.

3.2 Failure Analysis Using Synthetic RTN Data

To estimate the parameters of the measured RTN signal, it is important that they are validated because we cannot know the ground truth. Therefore, we conduct a Monte Carlo simulation experiment to evaluate the performance of proposed method.

3.2.1 Experimental Procedure

Amplitudes w and time constants τ are randomly sampled from the following distributions

$$w \sim \text{lognormal}(-1, 2) \text{ and} \quad (53)$$

$$\tau \sim \text{lognormal}(7, 1), \quad (54)$$

where $\text{lognormal}(w|\mu, \sigma)$ indicates log-normal distribution with the following probability density function,

$$p(w|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma x} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right). \quad (55)$$

In order to choose the distribution suitable for the following experiments, we examined existing studies though most of the analysis are limited to a single trap. One paper [6] reports that the amplitudes are well represented by log-normal distribution. On the other hand, it looks there is no consensus to the time constants distribution. In the following, based on [10], [11], we adopt log-normal distribution as the time constants model. We first generate a clean RTN signal by using w and τ generated from the distributions above. Then, Gaussian noise that simulates measurement noise is injected. Here, the standard variation of the noise is set to 0.01. We then compare the estimated parameters with those used for generating the test sequence. Note here that the comparison is non-trivial because the trap permutation can be arbitrary. For example, the parameters of trap #1 in Table 2 correspond to those of trap #3 in Table 3. We need a unified approach to match the estimated traps to those in the generation.

In this experiment, we solve this permutation problem by maximizing the match between the temporal trap sequences of the generated and estimated signals. For this purpose, we introduce the Rogers-Tanimoto coefficient [12] to evaluate the similarity of trap states \mathbf{x} and \mathbf{y} ,

$$D_{\text{Rogers-Tanimoto}}(\mathbf{x}, \mathbf{y}) = \frac{a_{00} + a_{11}}{a_{00} + a_{11} + 2(a_{01} + a_{10})}. \quad (56)$$

Here, a_{00} , a_{01} , a_{10} , and a_{11} are

$$a_{00} = \#(x_i = 0 \wedge y_i = 0), \quad (57)$$

$$a_{01} = \#(x_i = 0 \wedge y_i = 1), \quad (58)$$

$$a_{10} = \#(x_i = 1 \wedge y_i = 0), \text{ and} \quad (59)$$

$$a_{11} = \#(x_i = 1 \wedge y_i = 1), \quad (60)$$

where x_i and y_i are the temporal sequences of the trap states for time-step i . The coefficient increases when two traps are more similar. Then, the permutation that maximizes the above similarity is used for the evaluation. Once correspondence between traps is determined, we can define estimation errors corresponding to each trap.

3.2.2 Confidence Evaluation of Estimated Parameters

Because there is no way of knowing the ground truth of parameters, i.e., the amplitudes and time constants, a method

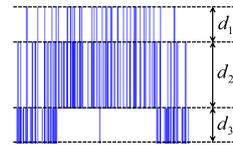


Fig. 6 Distances between states. The distances between each V_{TH} levels (d_1, d_2, d_3) should be sufficiently large to distinguish each states.

to evaluate the confidence of the estimated parameters is definitely required. For that purpose, we propose the following criteria to recognize the accuracy of the decomposed multi-trap parameters in the proposed method.

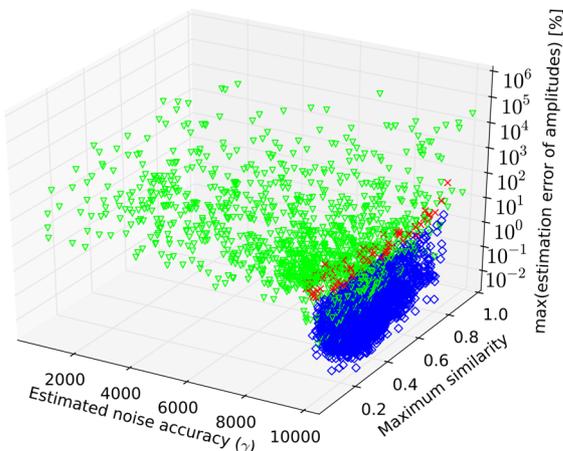
Estimated noise accuracy γ Good noise accuracy, which can be observed as γ (the inverse of the noise variance), indicates good agreement between the input and reconstructed signals. This means that the proposed method successfully decomposed the trap properties.

Similarities between estimated trap states When multiple traps exhibit similar temporal trapping-and-emitting sequences, it is difficult to estimate the parameters of each trap separately. In these input signals, confidence of the estimated result becomes low. Here, the similarities between estimated trap sequences can again be evaluated using the Rogers-Tanimoto coefficient. The maximum similarity is used as a criterion among possible $K(K+1)/2$ pairs for K traps.

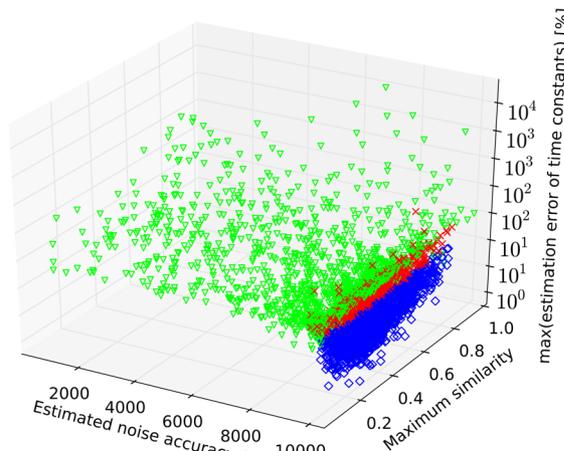
Difference in V_{TH} levels between states There is an ambiguity in the amplitude if the threshold voltage shift caused by some trap combinations is similar to that caused by another trap combination (Fig. 6). The histogram peaks formed by projection onto the threshold voltage axis should be clearly distinguished from each other. We calculate the reconstructed signal and then calculate a voltage histogram to investigate the distances between states. When there are K traps, the number of V_{TH} levels to be distinguished is 2^K . This means that the problem becomes inherently difficult as the number of states increases. The minimum distance is used as the criterion.

3.2.3 Experimental Result

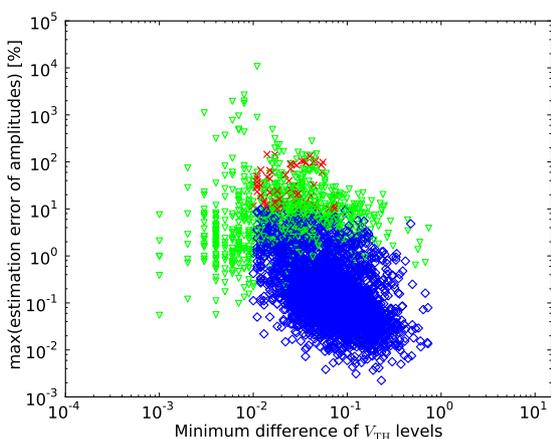
The results of the Monte Carlo experiment are shown in Figs. 7 and 8. Here, two traps are used and 5,000 samples are generated. The parameters for the prior distributions are set to $\alpha_\gamma = 10^5$, $\beta_\gamma = 10^{-1}$, $\lambda_{\alpha_w} = 1.0$, $\lambda_{\beta_w} = 0.1$, $\alpha_\pi = 1.0$, $\beta_\pi = 1.0$, $\lambda_{\alpha_{1/\tau}} = 10$, and $\lambda_{\beta_{1/\tau}} = 1$. Figures 7(a) and 8(a) show the maximum estimation error of the amplitudes and time constants (Z axis) versus the estimated noise accuracy γ and maximum similarity of the estimated trap states (X and Y axis), respectively. Figures 7(b) and 8(b) show the maximum estimation error of the amplitudes and time constants (Y axis) versus the minimum distance of V_{TH} levels (X axis), respectively. The blue diamonds represent estimated results that fall into the trust region and those for which the maximum estimation error of the amplitudes and



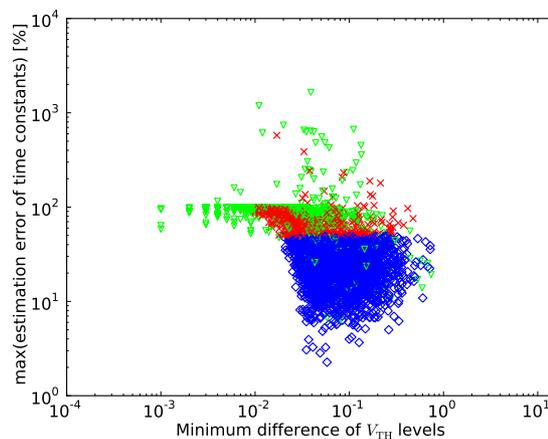
(a) Estimation accuracy of the amplitude versus the estimated noise accuracy γ and maximum similarity of trap states



(a) Estimation accuracy of the time constants versus the estimated noise accuracy γ and maximum similarity of trap states



(b) Estimation accuracy of the amplitude versus the minimum distance of the V_{TH} levels



(b) Estimation accuracy of the time constants versus the minimum distance of the V_{TH} levels

Fig. 7 Estimation accuracy of the amplitudes. Samples inside the trust region are indicated by red crosses and blue diamonds. Red crosses indicate that the maximum estimation error of the amplitudes is larger than 10%. Blue diamonds indicate that the proposed method successfully estimated the amplitude with a 10% accuracy. Samples outside the trust region are indicated by the green triangles.

Fig. 8 Estimation accuracy of the time constants. Samples inside the trust region are indicated by red crosses and blue diamonds. Red crosses indicate the maximum estimation error of the time constants is larger than 50%. Blue diamonds indicate the proposed method successfully estimated the time constant with a 50% accuracy. Samples outside the trust region are indicated by the green triangles.

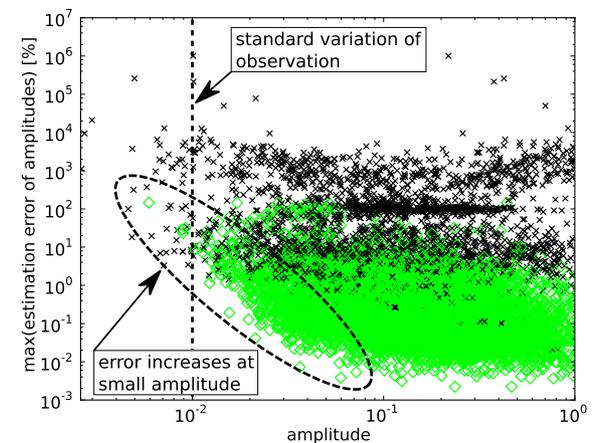
time constants is less than 10% and 50%, respectively. The red crosses indicate that estimated results fall into the trust region with at least one of the estimation errors of the amplitudes and time constants being greater than 10% and 50%, respectively. The green triangles indicate estimated results that fall outside of the trust region. The followings is a description of the trust region.

1. The estimated noise accuracy γ is greater than 9.8×10^3 .
2. The maximum similarity of estimated trap states is less than 0.9.
3. The minimum distance of the V_{TH} levels between estimated states is greater than 10^{-2} .

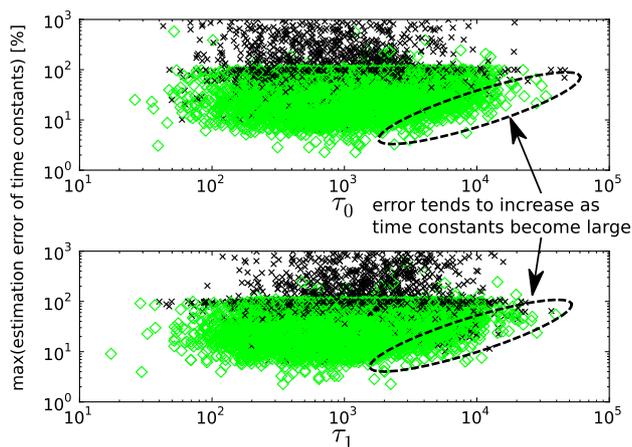
Because we assume that the standard variation of the measurement noise is 0.01, the estimated noise accuracy should be around 10^4 . By the same token, the distances between each state should be greater than the amplitude fluctuation

caused by noise. The maximum similarity reaches 1.0 if and only if two traps exhibit the same temporal trapping-and-emitting sequences. To exclude such cases, we introduce a condition for the similarity.

Figure 9 shows the relationship between amplitudes and time constants of generated sequences (X axis) and estimation accuracy (Y axis). The green diamonds represent estimated results that fall into the trust region while the black crosses represent estimated results that fall outside of the trust region. We can see that estimation accuracy becomes low as the amplitudes become small (less than 0.01) and the time constants become large (greater than 10k). In this Monte Carlo simulation experiment, 3,571 out of 5,000 samples fall into the trust region. Among those, 3,499 samples succeeded in estimating amplitudes within 10% accuracy. Similarly, 2,833 samples in the trust region succeeded



(a) The relationship between amplitudes and its estimation accuracy



(b) The relationship between time constants and its estimation accuracy

Fig. 9 The relationship between generation parameters and estimation accuracy. Green diamonds indicate the samples inside the trust region and black crosses indicate the samples outside of the trust region, respectively. The estimation error of amplitudes tend to increase as amplitude becomes small (a) while the estimation error of time constants tend to increase as time constants become large (b).

in estimating time constants within 50% accuracy. 2,823 samples simultaneously satisfied the above amplitudes and time constants bounds. On the other hand, relative errors become large for those samples having traps with small amplitudes. In our experiment, the estimation accuracy is judged by the maximum estimation error of all traps. Thus, traps with small amplitudes have a low estimation accuracy, whereas other parameters are estimated with good accuracy. Furthermore, estimating time constants is difficult owing to the weak sensitivity of the transition probabilities to the time constants. When τ is 100, for example, $\exp(-1/\tau)$ is about 0.990, which increases by only 0.005 when τ is increased to 200.

4. Experiments Using Measured RTN Data

We extract parameters from the measured threshold voltage shift of a pMOS transistor fabricated in a 65-nm pro-

Table 4 Estimated parameters from the measured RTN signal ($W/L=360/120$).

Trap	#1	#2
Amplitude [a.u.]	0.43×10^{-2}	1.12×10^{-2}
τ_0 [a.u.]	1.80×10^3	8.80×10^3
τ_1 [a.u.]	2.34×10^2	1.47×10^3

cess [13]. All parameters for the prior distribution remain fixed as those of the previous section. We have to know the ground truth of the noise variance of the measured RTN data so that we can apply the same trust region as we used in the previous section. To calculate the ground truth of the noise variance, measured signals unaffected by RTN are used. Then, measured signals with the RTN effects are scaled so that they have the same noise variance as the synthetic RTN signal in the previous section. Estimation results that fall outside the trust region defined in the previous section are omitted. These procedures are summarized as follows.

1. Find the trust region using the synthetic RTN data.
2. Apply a model estimation algorithm to the measured signal:
 - a. Calculate the noise variance using the measured data without the effects of RTN.
 - b. Scale the measured data that includes the RTN effects.
 - c. Omit the estimation results that fall outside the trust region.

The estimated results of 46 out of 128 devices fall into the trust region, one of them is shown in Fig. 10. The channel length L and width W are $W/L = 360 \text{ nm}/120 \text{ nm}$, respectively. The estimated amplitudes and time constants are listed in Table 4. The estimated noise accuracy γ is 1.27×10^4 . It takes about 15 minutes for estimating the parameters of a single device on Core i7 at 2.8 GHz.

From Fig. 10, we can see that the temporal sequences of two traps are clearly decomposed. The residue signal after subtraction of the reconstructed RTN shows that the effect of RTN is clearly removed. This means that the proposed method successfully captures the statistics of multi-trap RTN. Evaluation of the appropriateness of the parameters for existing research, although work is limited to the single observable trap case, is one topic of our future work. We carefully examined the estimation results and identified the questionable cases. The proposed estimation method seems to fail when the amplitude of threshold voltage shift is less than the amplitude of observation noise, or when time constants are larger than 10% of observation time period. Even if above conditions are not met, there are still some questionable results. We found that when the posterior distributions are multimodal, the proposed sampling procedure can be trapped in local optima, and thus fails to obtain the global optimum solution. We found that at the experiments on measured RTN data, the estimation of 15 out of 128 devices were failed due to above reason. Further improvement of the estimation algorithm using other sampling techniques,

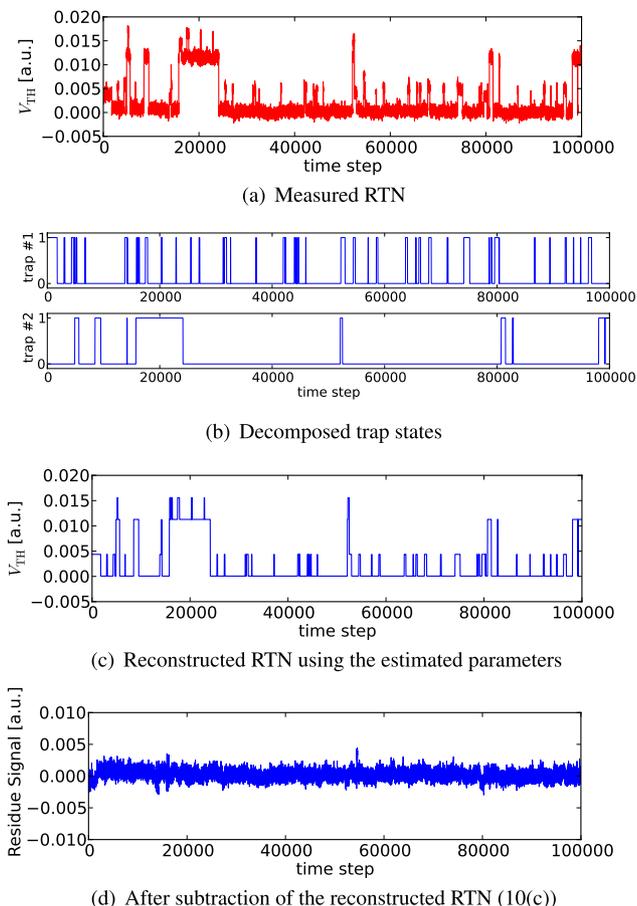


Fig. 10 Estimated trap states and reconstructed RTN signal ($W/L = 360/120$).

such as with parallel tempering, is also one topic of our future work.

5. Comparative Experiments with Existing Method

In this section, we compare our method with hidden Markov model (HMM) [14]. HMM is a popular and widely used method which is suitable to build a statistical model for time-domain sequences, such as voices. In the context of RTN analysis, HMM can be used to extract transition probabilities between discrete threshold voltage states. The main difference between the proposed method and HMM is that the proposed method takes into account the generation process of RTN while HMM does not. Large threshold voltage fluctuation is usually caused by combination of two or more trap activities. On the other hand, estimating the probability that a large threshold voltage shift occurs is difficult because such states that multi traps capture electrons will hardly be observed. In the following experiment, we show that extraction accuracy of such probability is improved over HMM by proposed method.

5.1 Experiment Setup

We show the estimation accuracy of the transition probab-

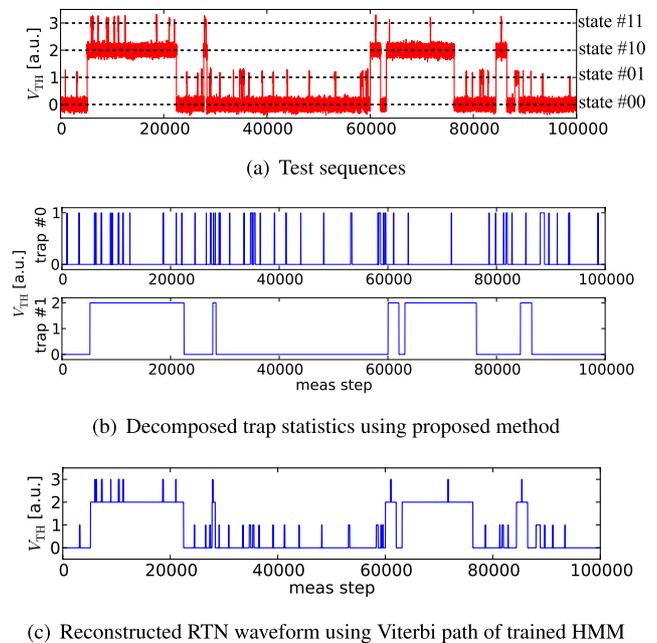


Fig. 11 Test sequences and estimated results by proposed method and HMM.

ity of proposed method and HMM using the artificially generated test sequence that is shown in Fig. 11(a). Here, we use the HMM of MATLAB toolkit [15]. Assuming that amplitudes are extracted accurately using other method such as TLP in advance to the transition probability estimation using HMM, the emission probabilities of each state are modeled as Gaussian distributions whose mean and variance are considered to be the ground truth. They are fixed during parameter update. Hence, in this experiment, HMM only estimates transition probabilities.

5.2 Results and Discussion

Figure 11(b) shows the decomposed trap states using the proposed method and Fig. 11(c) shows the reconstructed RTN waveform using Viterbi path of trained HMM [16]. The Viterbi path means the most probable sequence of hidden states for a given observation sequence. In the context of RTN analysis, the Viterbi path represent the most probable trapping-and-emitting sequence for a given threshold voltage sequence. The direct comparison between HMM and proposed method is difficult because HMM can only estimate transition probability matrix. Therefore, we create the transition probability matrix of the proposed method from the estimated time constants as shown in Fig. 12. For evaluation, we analyze the steady state of estimated Markov model. A steady state probability $\pi_i = (\pi_1, \dots, \pi_{2K})$ of Markov model is a solution of the following linear equation

$$A\pi = \pi, \quad (61)$$

where A is the transition probability matrix. The calculated steady state probabilities using estimated transition probability of HMM and the proposed method are listed in Ta-

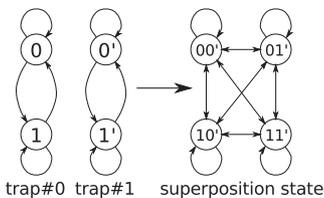


Fig. 12 Superposition state of two traps. For the comparison to HMM, we need to create transition probability matrix because proposed method outputs decomposed trap statistics.

Table 5 The steady state probabilities.

Method	State			
	00	01	10	11
Ground Truth	4.76e-1	2.39e-2	4.76e-1	2.39e-2
HMM	6.12e-1	3.68e-2	3.46e-1	5.53e-3
Error (HMM)	28.5%	53.8%	-27.3%	-76.9%
Proposed method	6.12e-1	3.23e-2	3.30e-1	1.72e-2
Error (proposed method)	30.2%	35.1%	-30.6%	-28.0%

ble 5. Note that HMM is given ground truth of amplitudes while proposed method is not. From Table 5, we can see that the proposed method accurately estimates the steady state probability of state 11 which gives large threshold voltage shift with only 28.0% error, while HMM exhibits 76.9% error. This difference comes from the fact that the proposed method can utilize the previous knowledge about generation process of RTN while HMM cannot.

6. Conclusion

In this study, we have proposed a statistical machine learning approach to simultaneously estimate the amplitude and time constants of each trap from measured threshold voltage fluctuation caused by RTN. The proposed method can handle interrelated parameters of multiple traps and thereby contributes to the construction of more accurate RTN models. The experiments using synthetic and measurement data showed that the proposed method successfully estimated the magnitudes of the threshold voltage shift. Further improvement of the estimation algorithm and application of the proposed method to a variety of MOS devices are the subject of future work.

Acknowledgment

This research is partially supported by a Grant-in-Aid for Scientific Research (B) 22360143 and JST CREST. Numerical computations were carried out on the super computer system, ACCMS, Kyoto University.

References

- [1] T.H. Morshed, M.V. Dunga, J. Zhang, D.D. Lu, A.M. Niknejad, and C. Hu, "Compact modeling of flicker noise variability in small size MOSFETs," *Tech. Dig. IEDM*, pp.1-4, Dec. 2009.
- [2] N. Tega, H. Miki, F. Pagette, D. Frank, A. Ray, M. Rooks, W. Haensch, and K. Torii, "Increasing threshold voltage variation

- due to random telegraph noise in FETs as gate lengths scale to 20 nm," *Symp. VLSI Tech. Dig. Tech. Papers*, pp.50-51, June 2009.
- [3] T. Nagumo, K. Takeuchi, S. Yokogawa, K. Imai, and Y. Hayashi, "New analysis methods for comprehensive understanding of random telegraph noise," *Tech. Dig. IEDM*, pp.1-4, Dec. 2009.
- [4] T. Nagumo, K. Takeuchi, T. Hase, and Y. Hayashi, "Statistical characterization of trap position, energy, amplitude and time constants by RTN measurement of multiple individual traps," *Tech. Dig. IEDM*, pp.28.3.1-28.3.4, Dec. 2010.
- [5] H. Miki, M. Yamaoka, N. Tega, Z. Ren, M. Kobayashi, C.P. D'Emic, Y. Zhu, D.J. Frank, M.A. Guillorn, D. Park, W. Haensch, and K. Torii, "Understanding short-term BTI behavior through comprehensive observation of gate-voltage dependence of RTN in highly scaled high-k metal-gate pFETs," *Symp. VLSI Tech. Dig. Tech. Papers*, pp.148-149, 2011.
- [6] S. Reanov and K.L. Shepard, "Random telegraph noise in 45-nm CMOS: Analysis using an on-chip test and measurement system," *Tech. Dig. IEDM*, pp.624-627, Dec. 2010.
- [7] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.PAMI-6, no.6, pp.721-741, Nov. 1984.
- [8] S. Moussaoui, D. Brie, A. Mohammad-Djafari, and C. Carteret, "Separation of non-negative mixture of non-negative sources using a Bayesian approach and MCMC sampling," *IEEE Trans. Signal Process.*, vol.54, no.11, pp.4133-4145, Nov. 2006.
- [9] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chemical Physics*, vol.21, no.6, pp.1087-1092, June 1953.
- [10] N. Tega, H. Miki, T. Osabe, A. Kotabe, K. Otsuga, H. Kurata, S. Kamohara, K. Tokami, Y. Ikeda, and R. Yamada, "Anomalously large threshold voltage fluctuation by complex random telegraph signal in floating gate flash memory," *Tech. Dig. IEDM*, pp.1-4, Dec. 2006.
- [11] M. Tanizawa, S. Ohbayashi, T. Okagaki, K. Sonoda, K. Eikyu, Y. Hirano, K. Ishikawa, O. Tsuchiya, and Y. Inoue, "Application of a statistical compact model for random telegraph noise to scaled-SRAM Vmin analysis," *Symp. VLSI Tech. Dig. Tech. Papers*, pp.95-96, June 2010.
- [12] M.M. Deza and E. Deza, *Encyclopedia of Distances*, Springer Berlin Heidelberg, 2009.
- [13] T. Sato, T. Kozaki, T. Uezono, H. Tsutsui, and H. Ochi, "A device array for efficient bias-temperature instability measurements," *Proc. ESSDERC*, pp.143-146, Sept. 2011.
- [14] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol.77, no.2, pp.257-286, Feb. 1989.
- [15] K. Murphy, "Hidden Markov Model (HMM) toolbox for Matlab," <http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>
- [16] G. Forney, Jr., "The viterbi algorithm," *Proc. IEEE*, vol.61, no.3, pp.268-278, 1973.



Hiromitsu Awano received his B.E. degree in Informatics from Kyoto University in 2010. Presently, he is a master course student at Department of Communications and Computer Engineering, Kyoto University. He is a student member of IPSJ.



Hiroshi Tsutsui received his B.E. degree in Electrical and Electronic Engineering and his master and Ph.D. degrees in Communications and Computer Engineering from Kyoto University in 2000, 2002, and 2005, respectively. He is currently an assistant professor in the Department of Communications and Computer Engineering, Kyoto University. His research interests include circuits and systems for image processing and VLSI design methodology. He is a member of IEEE, ACM, IPSJ, IEEJ, and IIEEJ.



Hiroyuki Ochi received the B.E., M.E., and Ph.D. degrees in Engineering from Kyoto University in 1989, 1991, and 1994, respectively. In 1994, he joined Department of Computer Engineering, Hiroshima City University as an associate professor. Since 2004, he has been an associate professor of Department of Communications and Computer Engineering, Kyoto University. His research interests include low-power/reliability-aware VLSI design and reconfigurable architectures. He is a member of IPSJ,

IEEE, and ACM.



Takashi Sato received B.E. and M.E. degrees from Waseda University, Tokyo, Japan, and a Ph.D. degree from Kyoto University, Kyoto, Japan. He was with Hitachi, Ltd., Tokyo, Japan, from 1991 to 2003, with Renesas Technology Corp., Tokyo, Japan, from 2003 to 2006, and with the Tokyo Institute of Technology, Yokohama, Japan. In 2009, he joined the Graduate School of Informatics, Kyoto University, Kyoto, Japan, where he is currently a professor. He was a visiting industrial fellow at the

University of California, Berkeley, from 1998 to 1999. His research interests include CAD for nanometer-scale LSI design, fabrication-aware design methodology, and performance optimization for variation tolerance. He received the Beatrice Winner Award at ISSCC 2000 and the Best Paper Award at ISQED 2003.