PAPER Special Section on Large Scale Algorithms for Learning and Optimization

Transfer Matrix Method for Instantaneous Spike Rate Estimation

Kazuho WATANABE^{†a)}, *Member*, Hiroyuki TANAKA^{††}, Keiji MIURA^{†††}, and Masato OKADA^{††††}, *Nonmembers*

SUMMARY The spike timings of neurons are irregular and are considered to be a one-dimensional point process. The Bayesian approach is generally used to estimate the time-dependent firing rate function from sequences of spike timings. It can also be used to estimate the firing rate from only a single sequence of spikes. However, the rate function has too many degrees of freedom in general, so approximation techniques are often used to carry out the Bayesian estimation. We applied the transfer matrix method, which efficiently computes the exact marginal distribution, to the estimation of the firing rate and developed an algorithm that enables the exact results to be obtained for the Bayesian framework. Using this estimation method, we investigated how the mismatch of the prior hyperparameter value affects the marginal distribution and the firing rate estimation.

key words: spike rate estimation, Bayesian estimation, transfer matrix method, smoothness prior

1. Introduction

A neuron interacts with other neurons through current pulses, which are called "spikes". Since the pulses of voltage have common forms between neurons, the activity of a neuron can be simplified into a sequence of spike timings. Since a spike sequence is affected by other neurons and/or external factors, it exhibits irregular behaviour and can be viewed as a stochastic process [1], [2]. There is assumed to be an underlying firing rate function which determines the probability of spike occurrence. This function is apparently used for conveying information between neurons.

Several methods have been developed for estimating the time-dependent firing rate function from spike sequence data. In the histogram method, a commonly used method [3]–[5], multiple spike sequences are recorded under the same conditions, and the firing rate is estimated from a histogram of the spike counts. However, it is generally difficult to obtain the same conditions in subsequent trials, and averaging over several trials can average the correlation with other neurons, thus reducing the informational value of this

Manuscript received December 5, 2008.

Manuscript revised February 4, 2009.

[†]The author is with the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma-shi, 630– 0192 Japan.

^{††}The author is with Toshiba Corporation, Kawasaki-shi, 212– 8582 Japan.

^{†††}The author is with JST, Kawaguchi-shi, 332–0012 Japan, Harvard University, Cambridge, MA 02138, USA and RIKEN, Wakoshi, 351–0198 Japan.

^{††††}The author is with the University of Tokyo, Kashiwa-shi, 277–8561 Japan and RIKEN, Wako-shi, 351–0198 Japan.

a) E-mail: wkazuho@is.naist.jp

DOI: 10.1587/transinf.E92.D.1362

measure.

One way to overcome this problem is to use a Bayesian framework to estimate the firing rate from a single sequence of spikes [6], [7]. With this approach, the generative process of the spike firing and the prior distribution of the firing rate are appropriately modelled, and the firing rate that has the largest posterior probability is selected. However, the firing rate is represented as a function of continuous time and has too many degrees of freedom to estimate. It is thus difficult to apply the Bayesian framework to such a large-scale model. Approximation methods, such as using the expectation-maximisation algorithm combined with Laplace approximation, have often been used [7]–[9].

We have applied the transfer matrix method to the Bayesian framework for use in estimating the firing rate and developed an algorithm that enables exact results to be obtained for the Bayesian framework. The transfer matrix method corresponds to belief propagation on a graph without any loops [10], [11]. It effectively computes the exact marginal distribution. The properties of the proposed method were investigated by calculating the marginal likelihood and estimation error of the firing rate. The rate function is estimated using a piecewise constant function over finely divided intervals of time called "bins". A method to estimate the number of bins from the Bayesian framework was also developed.

The rest of this paper is organised as follows. Section 2 describes the model for the firing rate estimation and its Bayesian framework. Section 3 presents the transfer matrix method for marginalisation. Section 4 describes the properties of the method as demonstrated by numerical experiments. Section 5 discusses the bin-width estimation and describes its experimentally demonstrated properties. Section 6 concludes the paper with a brief summary.

2. Firing Rate Estimation

2.1 Time-Dependent Poisson Process

Suppose we are given a single spike sequence $\{s\} = \{s_1, s_1, \dots, s_n\}$ observed in [0, T], where s_i is the time when a neuron fires and $0 < s_1 < s_2 < \dots < s_n < T$. The sequence is assumed to be generated in accordance with an underlying firing rate, $\lambda(t)$. Let $P(\{s\}|\lambda(t))$ be the probability distribution for spikes occurring in $\{s\}$ for a given $\lambda(t) \ge 0$. If spikes are assumed to occur independently at each instant of time, the probability density of the time-dependent Pois-

son process is given by

$$P(\{s\}|\lambda(t)) = \prod_{i=1}^{n} \lambda(s_i) \exp\left(-\int_0^T \lambda(t)dt\right),$$
(1)

where the exponential term is the survivor function, which represents the probability that no events occur in the interevent intervals. Equation (1) is a probability density function of {*s*} satisfying $s_1 < s_2 < \cdots < s_n$. The timedependent Poisson process is a special case of the ratefluctuating gamma process discussed by Koyama and Shinomoto [7].

2.2 Prior Distribution

Here, we assume the smoothness of the rate function [6], [7] and use a prior distribution of *N* variables $\{\lambda\} = \{\lambda_1, \dots, \lambda_N\}$ to reduce the number of degrees of freedom of the rate function. By dividing the observation period ($0 = t_1 < t_2 < \dots < t_{N+1} = T$), we get $\lambda_i = \lambda(t_i)$. The prior distribution is defined as

$$P(\lambda(t), \{\lambda\}) = \tilde{T}(\lambda(t))P(\{\lambda\}), \qquad (2)$$

where

$$P(\{\lambda\}) = \frac{1}{Z_0(\gamma, \sigma, \theta)} \exp\left\{-\frac{1}{2\gamma^2} \sum_{i=1}^N \frac{(\lambda_{i+1} - \lambda_i)^2}{\tau_i} -\frac{1}{2\sigma^2} \sum_{i=1}^N \tau_i (\lambda_i - \theta)^2\right\}$$
(3)

and $\tau_i = t_{i+1} - t_i$. In the following, we consider the intervals to be of equal length ($\tau = \tau_i$) for simplicity. The first term of the exponential part of $P(\{\lambda\})$ corresponds to the smoothness and the second term corresponds to the variation from the mean of the firing rate. Here, Z_0 is the normalization factor. The adjustable parameters (γ , σ , and θ) in the prior distribution are called "hyperparameters" [12]. The prior distribution $P(\{\lambda\})$ is similar to that used by Tanaka et al. [13] for the image restoration problem and provides a unified framework.

The \hat{T} in Eq. (2) describes a binding condition of the firing rate between λ_i and λ_{i+1} . We introduce a "bar-graph" model for use as a histogram representation of the firing rate. Let δ be the delta function and $\mathbf{1}_A$ be the indicator function of set A. The binding condition is given by

$$T(\lambda(t)) = \delta(\lambda(t) - \dot{\lambda}(t)), \tag{4}$$

where

$$\hat{\lambda}(t) = \sum_{i=1}^{N} \lambda_i \mathbf{1}_{[t_i \ t_{i+1})}(t).$$
(5)

Using Eqs. (2), (4), and (1) and marginalising over $\lambda(t)$, we get





$$P(\lbrace s \rbrace | \lbrace \lambda \rbrace) = \int P(\lbrace s \rbrace | \lambda(t)) P(\lambda(t) | \lbrace \lambda \rbrace) d\{\lambda(t)\}$$

=
$$\int P(\lbrace s \rbrace | \lambda(t)) \tilde{T}(\lambda(t)) d\{\lambda(t)\}$$

=
$$\prod_{i=1}^{N} (\lambda_i)^{\eta_i} \exp\left(-\sum_{i=1}^{N} \tau \lambda_i\right), \qquad (6)$$

where η_i is the number of spikes in the bin, $t_i \le t \le t_{i+1}$, as shown in Fig. 1.

2.3 Posterior Distribution

From Eqs. (3) and (6), Bayes' formula yields the posterior distribution of the *N* dimensional vector $\{\lambda\} = \{\lambda_1, \dots, \lambda_N\}$:

$$P(\{\lambda\}|\{s\}) = \frac{1}{Z} \prod_{i=1}^{N} (\lambda_i)^{\eta_i} \exp\left(-\sum_{i=1}^{N} (\tau \lambda_i + g_0(\lambda_{i+1}, \lambda_i))\right)$$
(7)

for which we have defined

$$g_0(\lambda_{i+1},\lambda_i) = -\frac{1}{2\gamma^2\tau}(\lambda_{i+1}-\lambda_i)^2 - \frac{\tau}{2\sigma^2}(\lambda_i-\theta)^2, \quad (8)$$

for $i = 1, 2, \dots, N$ and $\lambda_{N+1} = \lambda_N$. The normalisation constant,

$$Z = \int_0^\infty \prod_{i=1}^N d\lambda_i \prod_{i=1}^N (\lambda_i)^{\eta_i} \exp\left(-\sum_{i=1}^N (\tau \lambda_i + g_0(\lambda_{i+1}, \lambda_i))\right)$$
(9)

is called the "partition function".

We need to determine the restored rate from the posterior probability, Eq. (7). Here we use the posterior mean as the estimator:

$$\langle \lambda_i \rangle \equiv \int \prod_{j=1}^N d\lambda_j \,\lambda_i P\left(\{\lambda\}|\{s\}\right). \tag{10}$$

The posterior mean minimises the expected loss measured by the square error over the posterior distribution.

2.4 Hyperparameter Estimation

Since there are hyperparameters on which the right hand side (rhs) of Eq. (10) depends, we need to estimate them

from the spike sequence. We estimate these hyperparameters by using the maximum marginal likelihood method. In this method, the hyperparameters are determined by maximising the marginal likelihood function,

$$P(\lbrace s \rbrace) = \int_0^\infty \prod_{i=1}^N d\lambda_i P\left(\lbrace s \rbrace | \lbrace \lambda \rbrace\right) P\left(\lbrace \lambda \rbrace\right), \qquad (11)$$

which gives the probability of a particular spike sequence occurring. Equation (11) is proportional to Eq. (9) as a function of $\{s\}$. Therefore, we define the free energy as

$$F \equiv -\frac{1}{T} \log P(\{s\}) \tag{12}$$

The maximisation of the marginal likelihood function is equivalent to the minimisation of the free energy. Using Eqs. (1), (2), (3), and (4), we get

$$P(\lbrace s \rbrace) = \frac{\int_0^\infty \prod_i d\lambda_i \exp\left(\sum_{i=1}^N g(\lambda_{i+1}, \lambda_i)\right)}{\int_0^\infty \prod_i d\lambda_i \exp\left(\sum_{i=1}^N g_0(\lambda_{i+1}, \lambda_i)\right)},$$
(13)

where

$$g(\lambda_{i+1}, \lambda_i) = \eta_i \log \lambda_i - \tau \lambda_i + g_0(\lambda_{i+1}, \lambda_i), \qquad (14)$$

for $i = 1, 2, \dots, N - 1$ and

$$g(\lambda_{N+1},\lambda_N) = g_0(\lambda_{N+1},\lambda_N) = -\frac{\tau}{2\sigma^2}(\lambda_N - \theta)^2.$$
(15)

3. Transfer Matrix Method

The transfer matrix algorithm used to obtain the marginal distribution of the posterior distribution (Eq. (7)) and the marginal likelihood (Eq. (9)) is applicable if a stochastic process is a Markov chain.

The marginal distribution of the posterior distribution can be written as

$$P(\lambda_k|\{s\}) = \frac{1}{Z} \int_0^\infty \prod_{i=1}^{k-1} d\lambda_i \prod_{i=1}^{k-1} W(\lambda_{i+1}, \lambda_i)$$
$$\times \int_0^\infty \prod_{i=k+1}^N d\lambda_i \prod_{i=k}^{N-1} W(\lambda_{i+1}, \lambda_i),$$
(16)

where

$$W(\lambda_{i+1}, \lambda_i) = \exp(g(\lambda_{i+1}, \lambda_i)).$$

In Eq. (16), the rhs is divided into two integrals:

$$P(\lambda_k|\{s\}) = \frac{L_{k-1}^k(\lambda_k)R_{k+1}^k(\lambda_k)}{\int_0^\infty d\lambda_k L_{k-1}^k(\lambda_k)R_{k+1}^k(k)},$$

for which we have defined

$$L_{k-1}^{k}(\lambda_{k}) \equiv \int_{0}^{\infty} \prod_{i=1}^{k-1} d\lambda_{i} \prod_{i=1}^{k-1} W(\lambda_{i+1}, \lambda_{i}),$$

and

$$R_{k+1}^k(\lambda_k) \equiv \int_0^\infty \prod_{i=k+1}^N d\lambda_i \prod_{i=k}^{N-1} W(\lambda_{i+1}, \lambda_i)$$

Both integrals can be respectively reinterpreted as recursive relations:

$$\begin{aligned} L_{i-1}^{i}(\lambda_{i}) &= \int_{0}^{\infty} d\lambda_{i-1} L_{i-2}^{i-1}(\lambda_{i-1}) W(\lambda_{i}, \lambda_{i-1}), \\ L_{0}^{1}(\lambda_{1}) &= 1, \end{aligned}$$
 (17)

and

$$\begin{cases} R_{i+1}^{i}(\lambda_{i}) = \int_{0}^{\infty} d\lambda_{i+1} R_{i+2}^{i+1}(\lambda_{i+1}) W(\lambda_{i+1}, \lambda_{i}), \\ R_{N+1}^{N}(\lambda_{N}) = 1. \end{cases}$$
(18)

Equation (17) is the recursive relation in the forward direction, and Eq. (18) is in the backward direction. By defining the time complexity of one integral in Eq. (17) and one in Eq. (18) as M, we reduce the computational requirements from $O(M^N)$ to $O(M^2N)$ by this algorithm. More efficient algorithms based on the transfer matrix method have been proposed for particular cases [14], [15].

In the same way, the marginal likelihood function Eq. (13) can be reinterpreted as the recursive relation:

$$P(\{s\}) = \frac{\int_0^\infty d\lambda_N L_{N-1}^N(\lambda_N)}{\int_0^\infty d\lambda_N L_{N-1}'^N(\lambda_N)},$$

where

$$L_{i-1}^{\prime i}(\lambda_i) = \int_0^\infty d\lambda_{i-1} L_{i-2}^{\prime i-1}(\lambda_{i-1}) W^{\prime}(\lambda_i, \lambda_{i-1}),$$

$$W^{\prime}(\lambda_i, \lambda_{i-1}) = \exp\left(g_0(\lambda_i, \lambda_{i-1})\right).$$

4. Numerical Experiments

First, we investigated the averaged square error and free energy of our model. We define a prior distribution as

$$P_{s}(\{\lambda\}) = \frac{1}{Z_{0}(\gamma_{s}, \sigma_{s}, \theta_{s})} \exp\left\{-\frac{1}{2\gamma_{s}^{2}\tau} \sum_{i=1}^{N-1} (\lambda_{i+1} - \lambda_{i})^{2} - \frac{\tau}{2\sigma_{s}^{2}} \sum_{i=1}^{N} (\lambda_{i} - \theta_{s})^{2}\right\}.$$
(19)

This is the same distribution as Eq. (3) except for the hyperparameters γ_s , σ_s , and θ_s . We generated the firing rate following P_s which we call the "population prior distribution" in accordance with Tanaka et al. [13]. We refer to the generated rate as the "population firing rate" and to γ_s , σ_s , and θ_s as the "population hyperparameters". The spike sequences were generated from the population firing rate. The prior distributions $P(\{\lambda\})$ and $P_s(\{\lambda\})$ in Eqs. (3) and (19) correspond to the model and source prior distributions of Nishimori and Wong [16]. The population firing rate is referred to as the "source pixels" in the image restoration context [16].

This means that the piecewise constant function was

assumed to be the population firing rate function in our evaluation of the performance of the algorithm. This assumption is not too restrictive when the variation in the population rate function for a bin is sufficiently small.

The performance was evaluated using the mean square error per unit time as the metric:

$$[L] = \int d\{s\}d\{\lambda\}P(\{s\}|\{\lambda\})P_s(\{\lambda\})L$$
(20)

$$L = \frac{1}{T} \int_0^T dt \left(\lambda_s(t) - \lambda_m(t)\right)^2, \qquad (21)$$

where λ_s is the population firing rate given by $\{\lambda\}$ and λ_m is the estimated rate. The $\int d\{s\}$ and $\int d\lambda$ are the integrals over the spike sequence and the firing rate function, respectively. The averaged square error is also used in the image restoration context [13]. It can be proven that the averaged square error of the posterior mean estimation is minimised when the hyperparameters are equal to the population ones.

Figure 2 shows the averaged square error as a function of the hyperparameters. The population hyperparameters were set as $\gamma_s = 4$, $\sigma_s = 10$, and $\theta_s = 15$. Observation time *T* was 30, and the number of bins, *N*, was 150. In this figure, the gray level corresponds to the averaged square error. The transfer matrix algorithm was implemented using the double-exponential fast Gauss transform method [14]. Figure 2 (top) shows the averaged square error on the γ (horizontal axis)- σ (vertical axis) plane, Fig. 2 (middle) on the γ - θ plane, and Fig. 2 (bottom) on the σ - θ plane. The right side of each plane shows the averaged square error as a function of the horizontal axis. Note that the hyperparameters



Fig. 2 Averaged square error for hyperparameters γ , σ , and θ . Population hyperparameters were $\gamma_s = 4$, $\sigma_s = 10$, and $\theta_s = 15$.

for which the best performance was achieved were equal to the population hyperparameters. Therefore, we need to find hyperparameters close to the population ones to obtain good performance.

Figure 3 shows the averaged free energy, which is defined by

$$[F] = \int d\{s\}d\{\lambda\}P(\{s\}|\{\lambda\})P_s(\{\lambda\})F,$$

where F is defined by Eq. (12). The population hyperparameters were the same as for Fig. 2. Note that the hyperparameters that achieved the minimum value were equal to the population hyperparameters. Therefore, using the maximum marginal likelihood method to select hyperparameters implies that the estimated firing rate yields good perfor-



Fig. 3 Averaged free energy for hyperparameters γ , σ , and θ . Population hyperparameters were $\gamma_s = 4$, $\sigma_s = 10$, and $\theta_s = 15$ (T = 30, N = 150).



Fig.4 Averaged square error [*L*] with histogram method for different numbers of bins *N*.



Fig. 5 Scatter plots of three hyperparameters determined using free energy minimisation and squared error. Dotted curve shows averaged free energy. Population hyperparameters were $\gamma_s = 4$, $\sigma_s = 10$, $\theta_s = 15$, T = 30, and N = 300.

mance.

We also estimated the firing rate by using the histogram method, in which the rate function is determined by the ratio of the number of spikes within each bin to the bin size:

$$\lambda_i = \frac{\eta_i}{\tau}.\tag{22}$$

The averaged square error with the histogram method is shown in Fig. 4. Figures 4 and 2 show that the best performance with our model, about 7.3, is better than that with the histogram method, about 12.



Fig. 6 Example of population firing rate generated from Eq. (19), the rate estimated using the transfer matrix method, and rate estimated by histogram method.

Next, we investigated the performance of the posterior mean estimation and the maximum marginal likelihood method for each spike sequence. Figure 5 shows scatter plots of the square error as a function of hyperparameters γ (a), σ (b), and θ (c). The population hyperparameters are the same as for Fig.2. The estimated hyperparameters are distributed around the population hyperparameters. The Nelder-Mead algorithm was used for minimisation. It sequentially generates simplexes in the space of hyperparameters, for which the function values of F at the vertices are evaluated. The overall computational complexity is $O(M^2N)$ for the transfer matrix method, multiplied by the number of evaluations of F in the Nelder-Mead algorithm. Figure 6 shows examples of the population firing rate, the rate estimated using the transfer matrix method, and the rate estimated using the histogram method with the bin size that achieved the best performance. The parameters were $\gamma_s = 4$, $\sigma_s = 10, \theta_s = 15, T = 30$, and N = 300. We can see that the proposed method had better performance than the histogram method.

5. Bin Width Estimation

In the discussion above, the size of the histogram, τ , was fixed. Here it is considered a hyperparameter of the prior distribution and dealt with in the same way as γ , σ , and θ . We estimate the bin width by maximising the marginal likelihood. For simplicity, we divide the interval [0, T] into N bins with equal width and vary the number of bins instead of the width. The bin width and number of bins are related by $\tau = T/N$.

We examine the averaged free energy using the transfer matrix method. As an example of the simplest case, we assume that the population firing rate sampled from Eq. (19) has a relatively small number of bins, as there are in Fig. 7. This true number of bins is denoted by N_s . We fix the other hyperparameters (γ , σ , and θ) to the population values although we can simultaneously estimate all the hyperparameters in practice. Figure 8 shows the averaged free energy



Fig.7 Example of firing rate and spike sequence for $N_s = 10$.



Fig. 8 Averaged free energy for different numbers of bins for $N_s = 10$. Hyperparameters γ , σ , and θ were set to population values.



Fig. 9 Averaged free energy for different numbers of bins for $N_s = 30$. Hyperparameters γ , σ , and θ were set to population values.

for different numbers of bins. The population prior distribution is defined by $\gamma_s = 4$, $\sigma_s = 15$, $\theta_s = 15$, and $N_s = 10$. We can see that the averaged free energy takes a minimum at $N = N_s$. It also takes relatively small values at the multiples of N_s , $N = 20, 30, \cdots$. This means that the change points of the firing rate can be accurately estimated using free energy minimisation. The overall trend shows that the larger



Fig. 10 Averaged free energy for different numbers of bins for $N_s = 100$. Hyperparameters γ , σ , and θ were set to population values.

the number, the smaller the free energy. This is because the degrees of freedom increase with the number of bins.

Figures 9 and 10 show the averaged free energy for $N_s = 30$ and 100. In both cases, the free energy takes a minimum at $N = N_s$, demonstrating the practical applicability of bin-width estimation. We can see that the difference in the free energy between the optimal number, $N = N_s$, and the number around it decreases as N_s increases. This reflects the fact that, when N is large, the degrees of freedom are sufficiently large around N_s and the effect of a change in the number of bins is relatively small.

6. Conclusion

We have applied the transfer matrix method to the estimation of the time-dependent firing rate function from a single spike sequence, which enables exact results to be obtained for the Bayesian estimation. The prior distribution of the firing rate is described by a Gaussian model, and the binding condition is described by a bar-graph model. These models can be generalised to other models and binding conditions.

We evaluated the performance of the posterior mean estimation numerically by implementing the transfer matrix method. We demonstrated that the averaged square error is minimised when the model hyperparameters are equal to the population ones. We also demonstrated that the averaged free energy is minimised when they are equal to the population ones. These results imply that using the maximum marginal likelihood method to determine the hyperparameters and using the posterior mean estimation method to estimate the firing rate is an effective way to achieve accurate estimation. We also demonstrated that this combination had better performance than the histogram method.

Finally, we demonstrated that our unified framework using free energy minimisation also accurately estimates the bin width. This Bayesian treatment of bin-width estimation has generality and applicability to various statistical models.

References

[1] D.L. Snyder, Random Point Processes, Wiley, New York, 1975.

- [2] F. Rieke, D. Warland, R. de Ruyter van Steveninck, and W. Bialek, Spikes: Exploring the Neural Code, MIT Press, Cambridge, 1997.
- [3] H. Shimazaki and S. Shinomoto, "A method for selecting the bin size of a time histogram," Neural Comput., vol.19, no.6, pp.1503–1527, 2007.
- [4] S. Koyama and S. Shinomoto, "Histogram bin width selection for time-dependent Poisson processes," J. Phys. A, vol.37, pp.7255– 7265, 2004.
- [5] M. Rudemo, "Empirical choice of histograms and kernel density estimators," Scandinavian Journal of Statistics, vol.9, pp.65–78, 1982.
- [6] S. Koyama, T. Shimokawa, and S. Shinomoto, "Phase transitions in the estimation of event rate: A path integral analysis," J. Phys. A, vol.40, pp.F383–F390, 2007.
- [7] S. Koyama and S. Shinomoto, "Empirical Bayes interpretations of random point events," J. Phys. A, vol.38, pp.L531–L537, 2005.
- [8] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," J. Royal Statistical Society, vol.39-B, pp.1–38, 1977.
- [9] A.C. Smith and E.N. Brown, "Estimating a state-space model from point process observations," Neural Comput., vol.15, pp.965–991, 2003.
- [10] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, Palo Alto, CA, 1988.
- [11] Y. Kabashima and D. Saad, "Belief propagation vs. TAP for decoding corrupted messages," Europhys. Lett., vol.44, pp.668–674, 1998.
- [12] Z. Zhou, R.M. Leahy, and J. Qi, "Approximate maximum likelihood hyperparameter estimation for Gibbs priors," IEEE Trans. Image Process., vol.6, no.6, pp.844–861, 1997.
- [13] H. Tanaka, K. Miura, and M. Okada, "Image restoration with a truncated Gaussian model," J. Physical Society of Japan, vol.77, no.3, 034003-1-6, 2008.
- [14] M. Broadie and Y. Yamamoto, "A double-exponential fast Gauss transform algorithm for pricing discrete path-dependent options," Oper. Res., vol.53, no.5, pp.764–779, 2005.
- [15] E. Reiner, "Convolution methods for exotic options," Presentation at Columbia University, New York, 2000.
- [16] H. Nishimori and K.Y.M. Wong, "Statistical mechanics of image restoration and error-correcting codes," Phys. Rev. E, vol.60, no.1, pp.132–144, 1999.



Hiroyuki Tanaka was born in 1983. He received a B.Sc. degree from Tokyo Metropolitan University, Japan, in 2006 and a M.Sc. degree from the University of Tokyo, Japan, in 2008. He is currently a researcher in the Corporate Research & Development Center at Toshiba Corporation. His research interests include natural language processing, and its statistical applications.



Keiji Miura was born in 1978. He received a Ph.D. degree from Kyoto University, Japan, in 2006. He is currently a PRESTO researcher at the Japan Science and Technology Agency. His research interests include theoretical neuroscience.



Masato Okada received B.Sc. and M.Sc. degrees in physics and a Ph.D. degree in science, all from Osaka City University, Osaka, Japan, in 1985, 1987, and 1997, respectively. From 1987 to 1989, he worked at Mitsubishi Electric Corporation. From 1989 to 1991, he was a student in the Graduate School of Engineering Science, Osaka University, and from 1991 to 1996 he was a research associate at Osaka University. He was a researcher on the Kawato Dynamic Brain Project until 2001. He was a Deputy Head of the

Laboratory for Mathematical Neuroscience at the RIKEN Brain Science Institute, Saitama, Japan, and a PRESTO researcher on intelligent cooperation and control at the Japan Science and Technology Agency until 2004. He is currently a professor in the Graduate School of Frontier Science, University of Tokyo. His research interests include the computational aspects of neural networks and statistical mechanics for information processing.



Kazuho Watanabe was born in 1980. He received B.E., M.E. and Ph.D. degrees from the Tokyo Institute of Technology, Japan, in 2002, 2004 and 2006, respectively. From 2007 to 2008, he was a research associate in the Department of Complexity Science and Engineering at the University of Tokyo. He is currently an assistant professor at the Graduate School of Information Science, Nara Institute of Science and Technology. His research interests include statistical learning and its applications.