



Effect of Logistic Activation Function and Multiplicative Input Noise on DNN- k WTA Model

Wenhao Lu¹, Chi-Sing Leung^{1(✉)}, and John Sum²

¹ Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong

wenhaolu3-c@my.cityu.edu.hk, eeleungc@cityu.edu.hk

² Institute of Technology Management, National Chung Hsing University, Taichung, Taiwan

pfsun@nchu.edu.tw

Abstract. The dual neural network-based (DNN) k -winner-take-all (k WTA) model is one of the simplest analog neural network models for the k WTA process. This paper analyzes the behaviors of the DNN- k WTA model under these two imperfections. The two imperfections are, (1) the activation function of IO neurons is a logistic function rather than an ideal step function, and (2) there are multiplicative Gaussian noise in the inputs. With the two imperfections, the model may not be able to perform correctly. Hence it is important to estimate the probability of the imperfection model performing correctly. We first derive the equivalent activation function of IO neurons under the two imperfections. Next, we derive the sufficient conditions for the imperfect model to operate correctly. These results enable us to efficiently estimate the probability of the imperfect model generating correct outputs. Additionally, we derive a bound on the probability that the imperfect model generates the desired outcomes for inputs with a uniform distribution. Finally, we discuss how to generalize our findings to handle non-Gaussian multiplicative input noise.

Keywords: DNN- k WTA · Logistic activation function · Threshold logic units (tlus) · Multiplicative Input Noise

1 Introduction

The goal of the winner-take-all (WTA) process is to identify the largest number from a set of n numbers [1]. The WTA process has many applications, including sorting and statistical filtering [2, 3]. An extension of the WTA process is the k -winner-take-all (k WTA) process [4, 5], which aims to identify the k largest numbers from the set. From the dual neural network (DNN) concept, Hu and Wang [5] proposed a low complexity k WTA model, namely DNN- k WTA. This model contains n input-output (IO) neurons, one recurrent neuron, and only $2n + 1$ connections.

For ideal realization, the activation function of IO neurons should behave like a step function and there are no noise in the inputs. However, for circuit realization, the activation function often behaves like a logistic function [6, 7]. In addition, the operation of IO neurons may be affected by random drifts and thermal noise [8–10].

These two imperfections can affect the functional correctness. In [11, 12], Sum et al. and Feng et al. presented the analytical results of a noisy DNN- k WTA network, including the convergence and the performance degradation. However, Also, those results are based on the assumption that the noise are additive. In some situations, noise level are proportional to the signal levels. For instance, when we amplify the signal, the noise are amplified too. Hence, it is more suitable to use the multiplicative noise model to describe the behavior of the input noise [13, 14].

This paper analyzes the imperfect DNN- k WTA model with non-ideal activation function and existence of multiplicative input noise. We first assume that the multiplicative input noise are zero mean Gaussian distributed and then perform the analysis. Afterwards, we generalize our result to the non-Gaussian input noise. We derive an equivalent model to describe the behaviour of the imperfect DNN- k WTA model. From the equivalent model, we derive sufficient conditions to check whether the imperfect model can generate the desired results or not. We can use this condition to study the probability of the model generating correct outputs without simulating the neural dynamics. For uniformly distributed inputs, we derive a lower bound formula to estimate the probability that the imperfect mode can generate the correct outputs. Finally, we generalize our results to the non-Gaussian multiplicative input noise case.

This paper is organized as follows. Section 2 presents the background of the DNN- k WTA model. Section 3 studies the properties and performance of the DNN- k WTA models under the two imperfections. Section 4 extends the result to the non-Gaussian input noise case. Experimental results are shown in Sect. 5. Section 6 summarizes our results.

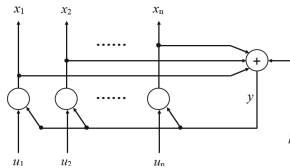


Fig. 1. Structure of a DNN- k WTA network.

2 Basic DNN- k WTA

Figure 1 illustrates the DNN- k WTA structure, which consists of a recurrent neuron and n input-output (IO) neurons. The state of the recurrent neuron

is denoted as $y(t)$. Each of the IO neurons has an external input, denoted as $\{u_1, \dots, u_n\}$. It of them is associated an output, denoted as x_i . All inputs u_i are distinct and range from 0 to 1. In the context of the DNN- k WTA model, the recurrent state $y(t)$ is governed by

$$\epsilon \frac{dy(t)}{dt} = \sum_{i=1}^n x_i(t) - k, \text{ where } x_i(t) = h(u_i - y(t)) \text{ and } h(\varphi) = \begin{cases} 1 & \text{if } \varphi \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

where ϵ is the characteristic time constant, which depends on the recurrent neuron's capacitance and resistance. In (1), $h(\cdot)$ denotes the activation function of IO neurons. In the original DNN- k WTA model, $h(\cdot)$ is an ideal step function. A nice property of the DNN- k WTA model is that its state converges to an equilibrium state in finite time. At the equilibrium state, only the IO neurons with the k largest inputs produce outputs of 1. All other neurons produce outputs of 0.

3 Logistic DNN- k WTA with Input Noise

3.1 DNN- k WTA Under Imperfection

In realization, the activation function of IO neurons often resembles a logistic function [6], and noise is unavoidable in analog circuits. This paper considers the coexistence of these two imperfections in the DNN- k WTA model. The first imperfection is that the activation function is a logistic function, given by

$$h_\alpha(\varphi) = \frac{1}{1 + e^{-\alpha\varphi}}, \quad (2)$$

where α is the gain factor. Also, there are multiplicative noise at the inputs of IO neurons. That is, the noisy inputs are given by

$$u_i + \varepsilon_i(t) u_i, \quad (3)$$

where $\varepsilon_i(t) u_i$ is the input noise for the i -th input. In this model, the noise level depends the normalized noise $\varepsilon_i(t)$, as well as input u_i . In this paper, we assume that $\varepsilon_i(t)$'s are Gaussian distributed with zero mean and variance of σ^2 .

With the two imperfections, the behaviour of the DNN- k WTA can be described as

$$\frac{dy(t)}{dt} = \sum_{i=1}^n \tilde{x}_i(t) - k, \quad (4)$$

$$\tilde{x}_i(t) = h_\alpha(u_i + u_i \varepsilon_i(t) - y(t)), \quad (5)$$

$$h_\alpha(\varphi'_i) = \frac{1}{1 + e^{-\alpha\varphi'_i}}, \text{ where } \varphi'_i = u_i + u_i \varepsilon_i(t) - y(t). \quad (6)$$

In the presence of noise in the inputs, the outputs $\tilde{x}_i(t)$ may change with time. Therefore, for the DNN- k WTA model with input noise, it is necessary to take

multiple measurements of the outputs of IO neurons to obtain the average output values as the neurons' outputs.

Figure 2 illustrates the effect of non-activation function and multiplicative Gaussian noise. In the first case, as shown in Fig. 2(a), When the gain factor is large enough and the noise variance is small, the recurrent state converges to around 0.5299 and thus the outputs of the network are correct. When the gain parameter α is reduced to 15, the recurrent state converges to 0.5145 and thus the outputs are incorrect, as shown in Fig. 2(b). If we increase the noise level to 0.2, the recurrent state converges to 0.5164 and thus the outputs are incorrect too, as shown in Fig. 2(c). Clearly, the gain parameter value α and the noise level σ^2 can affect the operational correctness.

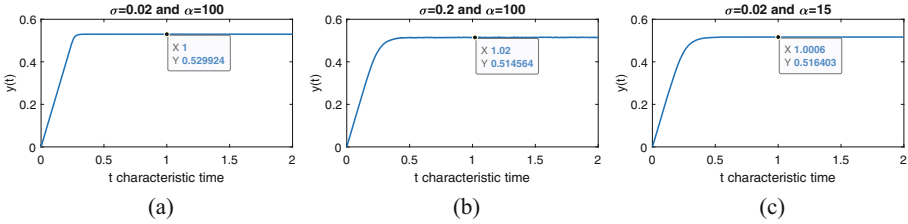


Fig. 2. The dynamics of the recurrent state in a DNN- k WTA with $n = 5$ and $k = 3$. The inputs are $\{u_1, \dots, u_5\} = \{0.54, 0.61, 0.52, 0.55, 0.51\}$. (a) Gain $\alpha = 100$ and noise level $\sigma = 0.02$. At the equilibrium, the recurrent state converges to 0.5299 and thus the outputs are $\{x_1, \dots, x_5\} = \{0.744, 0.999, 0.273, 0, 0.867, 0.137\}$. Clearly, only x_1 , x_2 and x_4 are greater than 0.5, and the outputs are correct. (b) Gain $\alpha = 100$ and noise level $\sigma = 0.2$. At the equilibrium, the recurrent state is 0.5145, the outputs are $\{x_1, \dots, x_5\} = \{0.592, 0.818, 0.528, 0, 0.587, 0.491\}$ and they are incorrect. (c) Gain $\alpha = 15$ and noise level $\sigma = 0.02$. At the equilibrium, the recurrent state is 0.5164. Thus, final outputs are $\{x_1, \dots, x_5\} = \{0.591, 0.806, 0.503, 0, 0.632, 0.485\}$ and they are incorrect.

3.2 Equivalent Model

This subsection derive a model to simulate the dynamic behaviour of the model under the two imperfections. We use the Haley's approximation for the Gaussian distribution [15].

Lemma 1. *Haley's approximation: A logistic function $\frac{1}{1+e^{-\rho z}}$ can be model by the distribution function of Gaussian random variables, given by*

$$\frac{1}{1+e^{-\rho z}} \approx \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} dv. \quad (7)$$

where $\rho = 1.702$.

From Lemma 1, the equivalent dynamics can be described by Theorem 1.

Theorem 1. *For the imperfect DNN-kWTA model with the two mentioned imperfections, we can use the following equations to describe its dynamic behaviour, given by*

$$\frac{dy(t)}{dt} = \sum_{i=1}^n \bar{x}_i(t) - k, \tag{8}$$

$$\bar{x}_i(t) = h_{\tilde{\alpha}_i}(u_i - y(t)), \tag{9}$$

$$h_{\tilde{\alpha}_i}(\varphi_i) = \frac{1}{1 + e^{-\tilde{\alpha}\varphi_i}}, \text{ where } \tilde{\alpha} = \frac{1}{\sqrt{\frac{1}{\alpha^2} + \frac{\sigma^2 u_i^2}{\rho^2}}} \text{ and } \varphi_i = u_i - y(t). \tag{10}$$

Proof: From (4), the update of the recurrent state can be written as

$$y(t + \delta) = y(t) + \delta \frac{dy(t)}{dt} = y(t) + \int_t^{t+\delta} \frac{dy(\tau)}{d\tau} d\tau = y(t) + \left(\sum_{i=1}^n \int_t^{t+\delta} \tilde{x}_i(\tau) d\tau - k\delta \right) \tag{11}$$

where δ is a small positive real number. The term $\int_t^{t+\delta} \tilde{x}_i(\tau) d\tau$ can be expressed as $\int_t^{t+\delta} \tilde{x}_i(\tau) d\tau = \lim_{M \rightarrow \infty} \zeta M \sum_{j=1}^M \frac{\tilde{x}_i(t+j\zeta)}{M}$, where $\zeta \times M = \delta$. We can further rewrite $\int_t^{t+\delta} \tilde{x}_i(\tau) d\tau$ as

$$\int_t^{t+\delta} \tilde{x}_i(\tau) d\tau = \delta \times (\text{mean of } \tilde{x}_i(t)) = \delta \times E[\tilde{x}_i(t)] = \delta \times \bar{x}_i(t) = \delta \times E[h_\alpha(\varphi'_i)],$$

where $\varphi'_i = u_i + \varepsilon_i(t) u_i - y(t)$ is the input of the i -th IO neuron. It contains the noise component $\varepsilon_i(t) u_i$. As $\varepsilon_i(t)$ is zero mean Gaussian distributed, we have

$$E[h_\alpha(\varphi'_i)] = \int_{-\infty}^{\infty} \frac{1}{1 + e^{-\alpha(\varphi_i + \varepsilon_i u_i)}} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\varepsilon_i^2}{2\sigma^2}} d\varepsilon_i,$$

where $\varphi_i = u_i - y(t)$. Furthermore, from Lemma 1,

$$\begin{aligned} E[h_\alpha(\varphi'_i)] &= \int_{-\infty}^{\infty} \int_{-\infty}^{\varphi_i} \frac{1}{\sqrt{2\pi\eta}} e^{-\frac{(v+u_i\varepsilon_i)^2}{2\eta}} \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\varepsilon_i^2}{2\sigma^2}} dv d\varepsilon_i \\ &= \frac{1}{\sqrt{2\pi\eta}} \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\varphi} \exp\left(-\frac{v^2}{2(\eta + \sigma^2 u_i^2)}\right) \\ &\quad \times \exp\left(-\frac{\left(\varepsilon_i + \frac{v u_i \sigma^2}{\sigma^2 u_i^2 + \eta}\right)^2}{2\eta\sigma^2 / (\sigma^2 u_i + \eta)}\right) dv d\varepsilon_i, \end{aligned} \tag{12}$$

where $\eta = \rho^2/\alpha^2$. Taking the integration with respect to ε_i and applying Lemma 1 again, we obtain

$$E[h_\alpha(\varphi'_i)] = \frac{1}{1 + e^{-\tilde{\alpha}_i s}} \triangleq \bar{x}_i(t) \triangleq h_{\tilde{\alpha}_i}(\varphi_i), \tag{13}$$

where $\varphi_i = u_i - y(t)$, $\eta = \rho^2 / \alpha^2$ and $\tilde{\alpha}_i = \frac{1}{\sqrt{\frac{1}{\alpha^2} + \frac{\sigma^2 u_i^2}{\rho^2}}}$. Equation (11) can be rewritten as

$$y(t + \delta) = y(t) + \delta \left(\sum_{i=1}^n \bar{x}_i(t) - k \right) = y(t) + \delta \frac{dy(t)}{dt}. \tag{14}$$

With (13) and (14), (4)–(6) can be written as

$$\frac{dy(t)}{dt} = \sum_{i=1}^n \bar{x}_i(t) - k, \tag{15}$$

$$\bar{x}_i(t) = h_{\tilde{\alpha}_i}(u_i - y(t)), \tag{16}$$

$$h_{\tilde{\alpha}_i}(\varphi_i) = \frac{1}{1 + e^{-\tilde{\alpha}_i \varphi_i}}, \tag{17}$$

where $\tilde{\alpha}_i = \frac{1}{\sqrt{\frac{1}{\alpha^2} + \frac{\sigma^2 u_i^2}{\rho^2}}}$ and $\varphi_i = u_i - y(t)$. The proof is completed. ■

It is important to note that we are not proposing a new model. Introducing the equivalent model, as stated in equations (15)–(17), helps us to analyze the properties of the imperfect model.

From Theorem 1, a convergence result of the DNN- k WTA network with the multiplicative Gaussian noise and non-ideal activation function is obtained. The result is presented in Theorem 2.

Theorem 2. *For the imperfect DNN- k WTA network, the recurrent state $y(t)$ converges to a unique equilibrium point.*

Proof: Recall that $\frac{dy}{dt} = \sum_{i=1}^n \frac{1}{1 + e^{-\tilde{\alpha}_i(u_i - y)}} - k$. For very large y , $\frac{dy}{dt} = -k < 0$. For very small y , $\frac{dy}{dt} = n - k > 0$. Furthermore, it is worth noting that $\frac{dy}{dt}$ is a strictly monotonically decreasing function of y . Therefore, there exists a unique equilibrium point y^* such that $\frac{dy}{dt}|_{y=y^*} = 0$. Additionally, we can obtain the following properties of the model:

$$\text{if } y(t) > y^*, \text{ then } \frac{dy}{dt} < 0, \text{ and if } y(t) < y^*, \text{ then } \frac{dy}{dt} > 0.$$

Suppose that at time t_o , the recurrent state $y(t_o)$ is greater than y^* . In this case, we have $\frac{dy}{dt} < 0$, and $y(t)$ decreases with time until it reaches y^* . On the other hand, if at time t_o the recurrent state $y(t_o)$ is less than y^* , then $y(t)$ also decreases with time until it reaches y^* . This completes the proof. ■

Since the output $\bar{x}_i(t)$ of the imperfect model is not strictly binary, we need to introduce new definitions for “winner” neurons and “loser” neurons.

Definition 1. *At the equilibrium, if $\bar{x}_i \geq 0.5$, then we call the i -th IO neuron is a winner. Otherwise, we call the i -th IO neuron is a loser.*

There are some relationship among equilibrium point y^* , winners and losers. The results are summarized in Theorem 3.

Theorem 3. Consider that the inputs are $\{u_1, \dots, u_n\}$. Denote the sorted inputs in the ascending order are $\{u_{\pi_1}, \dots, u_{\pi_n}\}$, where $\{u_{\pi_1}, u_{\pi_2}, \dots, u_{\pi_n}\}$ is the sorted index list. If $u_{\pi_{n-k}} < y^* \leq u_{\pi_{n-k+1}}$, then the imperfect model can generate correct outputs.

Proof: From (16) and (17), we know for a given y , if $u_i < u_{i'}$, then $h_{\bar{\alpha}_i}(u_i - y) < h_{\bar{\alpha}_{i'}}(u_{i'} - y)$. Also, $h_{\bar{\alpha}_i}(0) = 0.5$ and $h_{\bar{\alpha}_i}(u_i - y)$ is an increasing function of u_i . Thus, if $u_{\pi_{n-k}} < y^*$, then $\bar{x}_{\pi_1} < \dots < \bar{x}_{\pi_{n-k}} < 0.5$. Hence IO neuron π_1 to IO neuron π_{n-k} are losers. Similarly, if $u_{\pi_{n-k+1}} \geq y^*$, then $0.5 \geq \bar{x}_{\pi_{n-k+1}} > \dots > \bar{x}_{\pi_n}$. Hence IO neuron π_{n-k+1} to IO neuron π_n are winners. The proof is completed. ■

There is a common misconception that we can use Theorem 3 to study the probability of the imperfect model operating correctly. This approach involves simulating the neural dynamics for many sets of inputs, obtaining the equilibrium point y^* for each set, and then checking whether the imperfect model produces correct outputs or not. However, simulating the dynamics is quite time-consuming. Therefore, it is of interest to find a more efficient way to estimate the probability value of correct operation. The following theorem, based on the equivalent model, provides us with a convenient way to estimate the probability value without the need to simulate the neural dynamics.

Theorem 4. Denote the sorted inputs in the ascending order are $\{u_{\pi_1}, \dots, u_{\pi_n}\}$, where $\{u_{\pi_1}, u_{\pi_2}, \dots, u_{\pi_n}\}$ is the sorted index list. For the imperfect model,

$$\sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k}}} > k \quad \text{and} \quad \sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k+1}}} \leq k, \quad (18)$$

if and only if

$$u_{\pi_{n-k}} < y^* \leq u_{\pi_{n-k+1}}.$$

In addition, if

$$\sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k}}} > k \quad \text{and} \quad \sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k+1}}} \leq k,$$

then the model generates the desired outputs.

Proof: Denote $H(y) = \sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y) - k$. As $y \rightarrow \infty$, $H(y) = -k$. Also, as $y \rightarrow -\infty$, $H(y) = n - k$. Since $H(y)$ is a strictly monotonically decreasing function of y , we have $H(y) > 0$, $\forall y < y^*$, and $H(y) \leq 0$, $\forall y \geq y^*$. Hence $H(u_{\pi_{n-k}}) > 0$, if and only if, $u_{\pi_{n-k}} < y^*$. In addition, we have: $H(u_{\pi_{n-k+1}}) \leq 0$, if and only if, $u_{\pi_{n-k+1}} \geq y^*$. Furthermore,

$$\begin{aligned} & \text{if } \sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k}}} > k \quad \text{and} \quad \sum_{i=1}^n h_{\bar{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k+1}}} \leq k, \\ & \text{then } u_{\pi_{n-k}} < y^* \leq u_{\pi_{n-k+1}}. \end{aligned} \quad (19)$$

According to the result of Theorem 3, the condition “ $u_{\pi_{n-k}} < y^* \leq u_{\pi_{n-k+1}}$ ” implies that the model correctly identifies the winner and loser neurons for the given n numbers. ■

Theorem 4 provides us with an efficient way to estimate the probability value of the imperfect model correctly identifying the winner and loser neurons without the need to simulate the neural dynamics. To do so, we can consider many sets of inputs and sort the data for each input set. Then, we can compute the following expression:

$$\sum_{i=1}^n h_{\tilde{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k}}} - k \quad \text{and} \quad \sum_{i=1}^n h_{\tilde{\alpha}_i}(u_i - y)|_{y=u_{\pi_{n-k+1}}} - k$$

to determine whether the model can generate the correct output or not. It should be noticed that the aforementioned procedures are suitable for the data with any distribution. When the inputs are iid uniform random variables between zero and one, Theorem 5 provides us with a lower bound on the probability value.

Theorem 5. *If the inputs are iid uniform random variables with a range from zero to 1, the probability Prob(correct) that the imperfect model correctly identifies the winner and loser neurons can be expressed as:*

$$\text{Prob(correct)} \geq 1 - 2 \left(1 - \left(1 - \frac{2}{\tilde{\alpha}} \right)^{n-1} \left(1 + (n-1) \frac{2}{\tilde{\alpha}} \right) \right),$$

where $\tilde{\alpha} = 1/\sqrt{\frac{1}{\alpha^2} + \frac{\sigma^2}{\rho^2}}$.

Proof. Since the complete proof is lengthy, we only outline the flow of the proof here. Since the effect of zero-mean Gaussian noise is equivalent to decreasing the gain factor of the logistic function, we can use the flow of the proof in Theorem 4 of [7] to obtain our result. ■

Probability theory tells us that any non-uniformly distributed data can be mapped into a uniform distribution through histogram equalization. This mapping does not affect the ordering of the original non-uniform inputs. Therefore, we can apply Theorem 5 to handle non-uniformly distributed data.

4 Non-gaussian Multiplicative Input Noise

Although we focus on the multiplicative input noise with the Gaussian distribution, our analysis can be extended to cases where the multiplicative input noise has a non-Gaussian distribution. This technique is based on the idea of the Gaussian mixture model (GMM) [16]. We can use the GMM concept to approximate the density function of the normalized noise component $\varepsilon_i(t)$. In the context of GMM, the density function of $\varepsilon_i(t)$ can be represented as follows:

$$f(\varepsilon_i(t)) = \sum_{l=1}^L \frac{\Xi_l}{\sqrt{2\pi\zeta_l^2}} \exp \left\{ -\frac{(\varepsilon_i(t) - \mu_l)^2}{2\zeta_l^2} \right\}, \tag{20}$$

where μ_l is the mean of the l -th component, ς_l^2 is the variance of the l -th component, and Ξ_l is the weighting of the l -th component. Note that the sum of Ξ_l 's is equal to 1. By following the steps presented in Sect. 3, we can derive the equivalent dynamics for the case of non-Gaussian multiplicative input noise and non-ideal activation function. The equivalent dynamics are stated in the following theorem.

Theorem 6. *The equivalent dynamics for the non-Gaussian multiplicative input noise and non-ideal activation function are given by*

$$\frac{dy}{dt} = \sum_{i=1}^n \bar{x}_i(t) - k, \quad (21)$$

$$\bar{o}_i(t) = h_i(u_i - y(t)), \quad (22)$$

$$h_i(u_i - y(t)) = \sum_{l=1}^L \frac{\Xi_l}{1 + e^{-\tilde{\alpha}_{i,l}(u_i - y(t) + u_i \mu_i)}}, \quad (23)$$

$$\tilde{\alpha}_{i,l} = \frac{1}{\sqrt{\frac{1}{\alpha^2} + \frac{u_i^2 \varsigma_l^2}{\rho^2}}}. \quad (24)$$

Similar to the approach presented in Sect. 3, we can also develop an efficient method to estimate the probability of the model generating correct winner and loser neurons for the case of non-Gaussian multiplicative input noise and non-ideal activation function.

Theorem 7. *For the non-Gaussian multiplicative input noise and non-ideal activation function, if $\sum_{i=1}^n h_i(u_i - y(t))|_{y(t)=u_{\pi_{n-k}}} > k$ and $\sum_{i=1}^n h_i(u_i - y(t))|_{y(t)=u_{\pi_{n-k+1}}} \leq k$, then the model has the correct operation.*

5 Simulation Results

In Theorem 1 and Theorem 6, we introduce equivalent models to describe the behaviour of the imperfect DNN- k WTA model. Afterwards, based on the equivalent model, we propose the ways (Theorem 4, Theorem 5 and Theorem 7) to predict the performance of the imperfect model. The aim of this section is verified our results.

5.1 Effectiveness of Theorem 4

Three settings: $\{n = 6, k = 2, \alpha = 500\}$, $\{n = 11, k = 2\}$ and $\{n = 21, k = 5\}$ are considered. In this subsection, we consider that the inputs follow Beta distribution with $\text{Beta}_{c,d}(x) = \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} x^{c-1} (1-x)^{d-1}$, where $c = d = 2$, and $\Gamma(\cdot)$ denotes the well known Gamma function. To study the probability value of the imperfect model performing correctly, we generate 10,000 sets of inputs.

Time-varying multiplicative input noise “ $\varepsilon_i(t) u_i$ ” ’s are added into inputs, where $\varepsilon(t)$'s are zero-mean Gaussian distributed with variance of σ^2 . We consider

three gain values: $\{n = 6, k = 2, \alpha = 500\}$,
 $\{n = 11, k = 2, \alpha = 1000\}$ and $\{n = 21, k = 5, \alpha = 2500\}$.

When dealing with the non-uniform input case, we have two methods to measure the probability values of the imperfect model correctly identifying the winner and loser neurons. One way is to use the original neural dynamics stated in (4)–(6).

Another method is to use Theorem 4 to check the performance of the imperfect model. In this method, we only need to use (18) to determine whether the imperfect model can correctly identify the winner and loser neurons for each set of inputs. The results are shown in Fig. 3. It can be seen that the results obtained from Theorem 4 are quite close to the results obtained from the original neural dynamics over a wide range of noise levels and various settings.

For example, from Fig 3, for the case of $\{n = 6, k = 2, \alpha = 500, \sigma = 0.06309\}$, the two probability values from the two methods are 0.9416 and 0.09442, respectively.

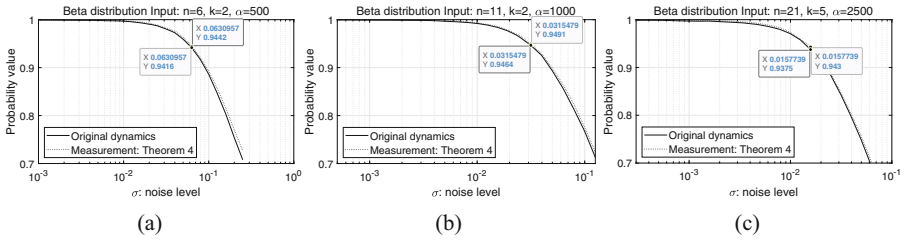


Fig. 3. The inputs are with the Beta distribution in $(0, 1)$, and the multiplicative input noise components are $\varepsilon_i(t)u_i$, where $\varepsilon(t)$'s are zero-mean Gaussian distributed with variance of σ^2 .

5.2 Effectiveness of Theorem 5

In this subsection, we study the effectiveness of Theorem 5. For uniform inputs, there is an additional method to estimate the performance. The simulation settings are similar to those of Sect. 5.2, except that the inputs are uniformly distributed.

When the inputs are uniformly distributed, we can use the lower bound from Theorem 5 to estimate the chance of identifying the correct winners and losers. The results are shown in Fig. 4. There are three methods to estimate the probability values. The first method is based on the original neural dynamics, which is quite time consuming. The second method is from Theorem 4, in which we should have input data sets. The last method involves using the lower bound from Theorem 5. The advantage of this method is that there is no need to perform the time-consuming simulation of the neural dynamics or require input data sets.

The results are in Fig 4. First, it can be seen that the probability values obtained from Theorem 4 are very close to those obtained from simulating the original neural dynamics. The probability values obtained from Theorem 5 are lower than the values obtained the original neural dynamics and Theorem 4. It is because Theorem 5 gives lower bounds on the probability values. But, the advantage of Theorem 5 is that there is no need to have input data sets.

We can use our result to know the noise tolerant level of the model. For example, for $\{n = 21, k = 5, \beta = 2500\}$ with the target probability value equal to 0.95, the input noise level σ should be less than 0.0157 from Theorem 4, while the result of the low bound tells us that the input noise level σ should be less than 0.00995.

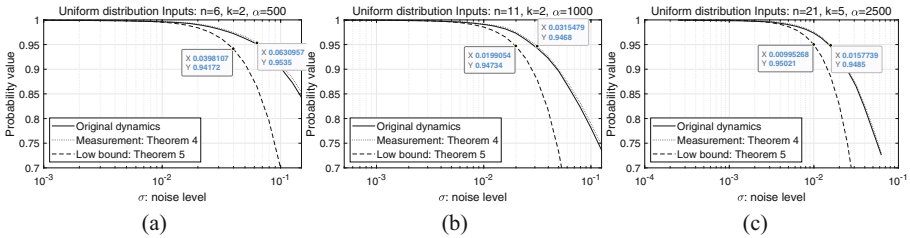


Fig. 4. The inputs are uniformly distributed in $(0, 1)$, and the multiplicative input noise components are $\varepsilon_i(t)u_i$, where $\varepsilon(t)$'s are zero-mean Gaussian distributed with variance of σ^2 .

5.3 Effectiveness of Theorem 7

We can Theorem 7 to predict the performance of the model for non-Gaussian distributed multiplicative input noise. To study the performance in this case, we consider that the inputs follow a uniform distribution with a range of $(0, 1)$ in this subsection. We generated 10,000 sets of inputs for this purpose.

We consider that **multiplicative input noise** are “ $\varepsilon_i(t)u_i$ ”’s, where $\varepsilon_i(t)$'s are uniformly distributed in the range $[-\Delta/2, \Delta/2]$. We chose to use a uniform distribution to demonstrate that the GMM concept is capable of handling non-bell-shaped distributions. This is because the uniform distribution has a rectangular shape, which is significantly different from the Gaussian distribution.

In the simulation, for each noise level, we build a GMM with 11 components. We consider three settings: $\{n = 6, k = 2, \alpha = 500\}$, $\{n = 11, k = 2, \alpha = 1000\}$ and $\{n = 21, k = 5, \alpha = 2000\}$.

To validate the effectiveness of Theorem 7, we also use the original neural dynamics to estimate the probability of the model having the correct operation. It should be noticed that this simulation method is quite time consuming. The results are shown in Fig. 5. From the figure, the result of Theorem 7 is very close to that of simulating the original neural dynamics.

Again, we can use Theorem 7 to predict the tolerant level for input noise. For example, for $\{n = 6, k = 2, \alpha = 500\}$ with the target probability value equal to 0.95, Theorem 7 tells us that the input noise range δ should be less than 0.1999.

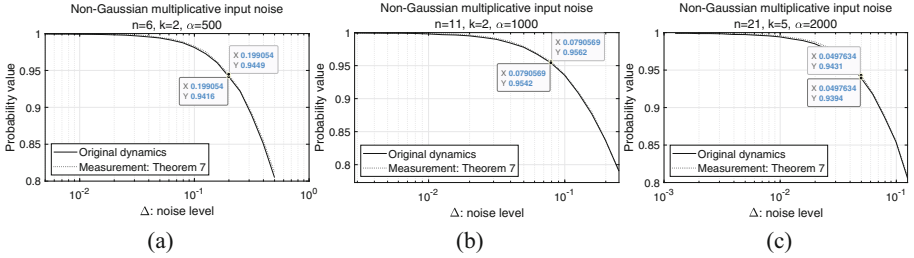


Fig. 5. The inputs are uniformly distributed in $(0,1)$, and the multiplicative input noise components are $\varepsilon_i(t)u_i$, where $\varepsilon(t)$'s are zero-mean uniformly distributed noise in the range of $[-\Delta, -\Delta]$.

6 Conclusion

This paper presented an analysis of the DNN- k WTA model with two imperfections, namely, multiplicative input noise and non-ideal activation in IO neurons. We first developed an equivalent model to describe the dynamics of the imperfect DNN- k WTA model. It should be aware that the equivalent model is introduced for studying behaviour of the imperfect DNN- k WTA model and that it is not a new model. Using the equivalent model, we derive sufficient conditions for checking whether the imperfect model can correctly identify the winner and loser neurons. For uniform-distributed inputs, we provide a formula to estimate the lower bound on the probability value of the model with correct operation. Lastly, we extend our results to handle the non-Gaussian multiplicative input noise case. We validate our theoretical results through various simulations.

References

1. Touretzky, S.: Winner-take-all networks of $O(n)$ complexity. In: Advances in Neural Information Processing Systems, vol. 1, pp. 703–711. Morgan Kaufmann (1989)
2. Kwon, T.M., Zervakis, M.: KWTA networks and their applications. *Multidimension. Syst. Signal Process.* **6**(4), 333–346 (1995)
3. Narkiewicz, J.D., Burleson, W.P.: Rank-order filtering algorithms: a comparison of VLSI implementations. In: the 1993 IEEE International Symposium on Circuits and Systems, pp. 1941–1944. IEEE (1993)
4. Sum, J.P., Leung, C.S., Tam, P.K., Young, G.H., Kan, W.K., Chan, L.W.: Analysis for a class of winner-take-all model. *IEEE Trans. Neural Netw.* **10**(1), 64–71 (1999)

5. Hu, X., Wang, J.: An improved dual neural network for solving a class of quadratic programming problems and its k -winners-take-all application. *IEEE Trans. Neural Netw.* **19**(12), 2022–2031 (2008)
6. Moscovici, A.: *High Speed A/D Converters: Understanding Data Converters Through SPICE*, vol. 601. Springer Science & Business Media, Berlin (2001)
7. Feng, R., Leung, C.S., Sum, J., Xiao, Y.: Properties and performance of imperfect dual neural network-based k WTA networks. *IEEE Trans. Neural Netw. Learn. Syst.* **26**(9), 2188–2193 (2014)
8. Redouté, J.M., Steyaert, M.: Measurement of EMI induced input offset voltage of an operational amplifier. *Electron. Lett.* **43**(20), 1088–1090 (2007)
9. Kuang, X., Wang, T., Fan, F.: The design of low noise chopper operational amplifier with inverter. In: 2015 IEEE 16th International Conference on Communication Technology (ICCT), pp. 568–571. IEEE (2015)
10. Lee, P.: Low noise amplifier selection guide for optimal noise performance. Analog Devices Application Note, AN-940 (2009)
11. Feng, R., Leung, C.S., Sum, J.: Robustness analysis on dual neural network-based k WTA with input noise. *IEEE Trans. Neural Netw. Learn. Syst.* **29**(4), 1082–1094 (2017)
12. Sum, J., Leung, C.S., Ho, K.I.J.: On Wang k WTA with input noise, output node stochastic, and recurrent state noise. *IEEE Trans. Neural Netw. Learn. Syst.* **29**(9), 4212–4222 (2017)
13. Semenova, N., et al.: Fundamental aspects of noise in analog-hardware neural networks. *Chaos: Interdisc. J. Nonlinear Sci.* **29**(10), 103128 (2019)
14. Kariyappa, S., et al.: Noise-resilient DNN: tolerating noise in PCM-based AI accelerators via noise-aware training. *IEEE Trans. Electron Devices* **68**(9), 4356–4362 (2021)
15. Haley, D.C.: Estimation of the dosage mortality relationship when the dose is subject to error. STANFORD UNIV CA APPLIED MATHEMATICS AND STATISTICS LABS, Technical report (1952)
16. Radev, S.T., Mertens, U.K., Voss, A., Ardizzone, L., Kothe, U.: BayesFlow: learning complex stochastic models with invertible neural network. *IEEE Trans. Neural Netw. Learn. Syst.* **33**(4), 1452–1466 (2020)