TOWARDS A UNIFIED MODEL FOR THE RETINA
Static vs Dynamic Integrate and Fire Models

Pedro Tomás, João Martins and Leonel Sousa
INESC-ID / IST TU Lisbon, Rua Alves Redol 9, 1000-029, Lisboa, Portugal

Keywords: Retina Models, Stochastic Leaky Integrate and Fire, Poisson-based model, Dynamic model.

Abstract: Many models have been proposed to describe the visual processing mechanisms in the retina. The spike generation mechanism of the models is typically performed by a Poisson process. Alternatively, a more realistic approach can be used by implementing an integrate and fire mechanism. In this paper we show that the Stochastic Leaky Integrate and Fire (SLIF) model is equivalent to a non-linear Poisson-based model. Furthermore, it proposes a dynamic model for the retina visual processing path, achieved through modulations. For estimating this model a two-step approach is proposed: *i)* an initial estimation is computed by using a spike-triggered analysis, and *ii)* the likelihood of the spike train is maximised by gradient ascent.

1 INTRODUCTION

Vision is a fundamental sense in one's everyday life that gained even more relevance in the modern society; most of the information, art and entertainment relies on it. A continuously growing number of research groups have been dedicating their efforts to help visually impaired people by developing visual prostheses capable of conveying some kind of vision; a endeavour which demands the development of accurate and reliable retina models.

Retinal ganglion cells respond to visual stimuli by eliciting spikes whenever the inner-membrane voltage potential surpasses a given threshold. This response, $y(t)$, is characterised by the time spikes occurred; it can be mathematically represented as a sum of Dirac functions $\delta(t-t_k)$ centred at the time instants t_k, when the inner-membrane voltage potential surpassed the threshold: $y(t)=\sum\delta(t-t_k)$.

Several retina models, based mainly on general neuron models, have been proposed. Two of the most typical are the Poisson based model (Chichilnisky, 2001) and the Stochastic Leaky Integrate and Fire (SLIF) (Paninski, 2006) model. However, in this paper it is shown that, under some constraints, they are equivalent. Moreover, a dynamic model is proposed based on developed mathematical machinery used to prove the equivalence between these two models. This dynamic model modulates its output not only by the stimulus characteristics but also by the recent spike firing history. To estimate this model's parameters several other issues are addressed, such as the tuning and initialisation of this model, where it is used spike-triggered analysis. At the end some experimental results are provided.

This paper is organised as follows. Section 2 shows the equivalence between the SLIF model and a Poisson based model. Section 3 presents the proposed dynamic model and describes its tuning. Some experimental results are drawn in Section 4, and section 5 concludes the paper.

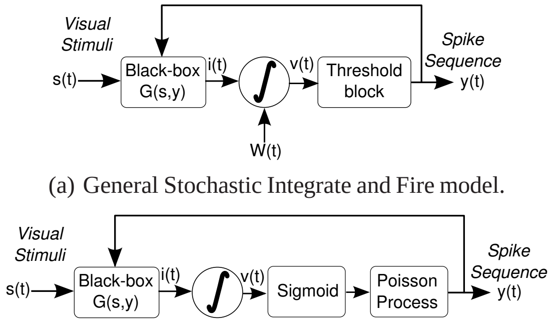
2 STOCHASTIC INTEGRATE AND FIRE MODEL

A typical approximation for the Hodgkin and Huxley neuron model assumes a leaky integrate and fire model. By adding a noise component to the model, it is possible to simulate the variability of real neurons (Keat et al., 2001). For a linear first order integrate and fire mechanism, the SLIF model is described by the stochastic differential equation (SDE):

$$dv(t)=-\frac{1}{\tau}v(t)dt+i(t)dt+\sigma\xi(t) \qquad (1)$$

where τ is a constant variable, $\xi(t)$ is standard white-noise and σ is a multiplicative term which defines the power of the noise source. The above SDE is valid until the subthreshold potential surpasses a given threshold V_{th}. Whenever this happens a spike is fired and the neuron enters a refractory period where no spike can be fired. Accordingly with the SLIF model, at the end of this period, which typically lasts a couple of

Tomás P., Martins J. and Sousa L. (2008).
TOWARDS A UNIFIED MODEL FOR THE RETINA - Static vs Dynamic Integrate and Fire Models.
In *Proceedings of the First International Conference on Bio-inspired Systems and Signal Processing*, pages 528-533
DOI: 10.5220/0001067905280533



(a) General Stochastic Integrate and Fire model.

(b) Generalised Poisson model.

Figure 1: Changing the noise source in Integrate and Fire models.

milliseconds, the neuron is reset to a state where the subthreshold potential is v_0. Considering that the last spike was fired at time $t = t_0$, the solution to (1) is[1]:

$$v(t) = h_{IF}(t - t_0) * \Big[i(t) + v_0\delta(t - t_0) + W(t)\Big] \quad \text{(2a)}$$

$$W(t) \sim N(0, \sigma_t) \qquad , \sigma_t \equiv \sigma(t) \quad \text{(2b)}$$

where $*$ is the convolution operator and $h_{IF}(t)$ is the transfer function of a low pass filter with a pole in $\frac{1}{\tau}$:

$$h_{IF}(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}} H(t) \quad (3)$$

where $H(t)$ is the Heaviside step function.

2.1 Equivalence between the Slif Model and the Poisson Based Model

Let us consider a SLIF model as shown in Figure 1(a), where $s(t)$ represents the visual stimulus, $y(t)$ is the spike response, and $G(s,y)$ is an arbitrary function that transforms the pair $\{$stimulus, spike response sequence$\}$ into the SLIF input $i(t)$. The integrator block in the figure represents the linear filter $h_{IF}(t)$.

Under the SLIF model assumption, the noise source can be moved to the integrator output by modifying its statistics: $v(t) = u(t) + W'(t) \sim N(u(t), \sigma_t^2)$. Considering that after the refrationary period the subthreshold potential v follows a Gaussian distribution with standard deviation σ_0, and noting that the convolution is a linear operator, the output noise source remains Gaussian, with standard deviation:

$$\sigma_t^2 = H(t - t_0)\left[\sigma^2 \frac{\tau}{2} + \left(\sigma_0^2 - \sigma^2 \frac{\tau}{2}\right) e^{-2\frac{t - t_0}{\tau}}\right] \quad (4)$$

where t_0 represents the time instant of the last spike and σ is the standard deviation of the noise during the integration period, i.e. the Interspike Interval (ISI).

Therefore, the probability for eliciting a spike at a given time instant t is given by the probability for the potential $v(t) = u(t) + W'(t)$ to surpass V_{th}:

$$\begin{aligned} P\Big(y(t) = 1\Big) &= P\Big(u(t) + W'(t) \geq V_{th}\Big) = \\ &= P\Big(W'(t) \geq V_{th} - u(t)\Big) = \\ &= 1 - \mathrm{N_{cdf}}(V_{th} - u(t)|0, \sigma_t^2) = \\ &= \mathrm{N_{cdf}}(u(t)|V_{th}, \sigma_t^2) \end{aligned} \quad (5)$$

where $\mathrm{N_{cdf}}(x|\mu, \sigma_t^2)$ is the normal cumulative distribution function with mean μ and variance σ_t^2, evaluated at point x. The function $\mathrm{N_{cdf}}$ defines a sigmoid where V_{th} controls the translation (centre of the sigmoid) and σ_t controls the expansion of the sigmoid.

Since the spiking probability of the SLIF model is given by a sigmoid function, the model in Figure 1(a) is equivalent to the Poisson based model in Figure 1(b), where the shape of the sigmoid depends on the noise statistic during the refractionary and integration periods; three cases can be considered:

(1) if $\sigma_0^2 = \sigma^2 \frac{\tau}{2}$, the noise variance throughout the integration period is always the same;

(2) if $\sigma_0^2 < \sigma^2 \frac{\tau}{2}$, the noise variance increases exponentially from σ_0^2, just after the refractory period, converging to $\sigma^2 \frac{\tau}{2}$;

(3) if $\sigma_0^2 > \sigma^2 \frac{\tau}{2}$, the noise variance decreases exponentially from σ_0^2, just after the refractory period, converging to $\sigma^2 \frac{\tau}{2}$;

Equivalently, for a neuron described by the Poisson spike generation process presented in Figure 1(b), the change in noise variance is translated into a varying slope of the sigmoid – except in case (1). In case (2)/(3), the model variability increases/decreases as time from last spike progresses.

However it is worth to notice that this is not a true Poisson model, as the firing of a spike at a time instant depends on the recent spiking history. Since the integrator is reset whenever a new spike is fired, this is true even without the feedback path.

The above conversion thus shows that integrate-and-fire (IF) models improve the precision of neuron models by adding three important functions to the typical Poisson based model: *i)* a natural refractionary period given by the time for the integrator to recharge and fire a second spike; *ii)* a feedback mechanism; and *iii)* a sigmoid-like nonlinearity. While the feedback mechanism is not absolutely required to achieve a reasonable precision – see (Capela et al., 2007) –, the modelling of the feedback mechanism with a sufficiently variable filter allows for a considerable improvement on the precision of the model – see (Tomás and Sousa, 2007).

[1] Notice that (1) is similar to the Langevin's SDE.

3 DYNAMIC MODEL

The typical modelling of neurons represents $G(s,y)$ in Figure 1 by a linear system dependent on the input stimuli $s(t)$ and on spike history $y(t)$:

$$G(s(t),y(t)) = \underbrace{(h_f * s)(t)}_{i_f(t)} + \underbrace{(h_b * y)(t)}_{i_b(t)} \tag{6}$$

where $h_f(t)$ and $h_b(t)$ represent the feedforward and feedback linear filters, respectively.

In some cases, driven by the necessity to model the contrast adaptation mechanisms existent in the retina, non-linear functions are included in the description of $G(s,y)$ (Baccus and Meister, 2002). These non-linearities can be, for instance, approximated by a Taylor series:

$$G(s(t),y(t)) = f_f(i_f(t)) + f_b(i_b(t)) \tag{7a}$$

$$f_x(y) = \sum_k a_k y^k \tag{7b}$$

Nevertheless, the inclusion of dynamics in neuron model is usually avoided, eventhough recent research suggest that they are required for effectively modelling the precise timing of neurons (Gerstner et al., 2006). This is particularly verified in IF neuron models where its natural non-linearities introduce local minimums, making the training process harder. In order to model the temporal dynamics of the retina circuitry, a model must be constructed where the transfer function changes with time, for instance by means of a modulation process. Typically, neural dynamics are considered only as a function of the stimuli – e.g. motion detection (Bialek and van Steveninck, 2005). However, as referred in section 2, if the precision of neuron models is improved by the inclusion of a feedback path, one should also consider the spike history for modelling neural dynamics.

The proposed model is divided into two parts: a static part and a dynamic part. Both parts have dependencies on the stimuli and the spike history. The static part is composed by a *static forward* filter $h_{sf}(t)$ and a *static backward* filter $h_{sb}(t)$:

$$i_{sf}(t) = (h_{sf} * s)(t) \tag{8a}$$

$$i_{sb}(t) = (h_{sb} * y)(t) \tag{8b}$$

The dynamic part is composed by a set of *dynamic forward* linear filters $h_{df}^{(k)}(t)$ and *dynamic backward* linear filters $h_{db}^{(k)}(t)$, whose amplitude is dynamically controlled by linear functions dependent of both the stimulus and the spike history:

$$i_{df}^{(k)}(t) = c_{df}^{(k)}(t)(h_{df}^{(k)} * s)(t) \tag{9a}$$

$$i_{db}^{(k)}(t) = c_{db}^{(k)}(t)(h_{db}^{(k)} * y)(t) \tag{9b}$$

$$c_{df}^{(k)}(t) = \left[(h_{fdf}^{(k)} * s)(t) + (h_{fdb}^{(k)} * y)(t)\right] \tag{9c}$$

$$c_{db}^{(k)}(t) = \left[(h_{bdf}^{(k)} * s)(t) + (h_{bdb}^{(k)} * y)(t)\right] \tag{9d}$$

where $h_x^{(k)}(t)$ represents the linear filter x for component k and $i_{df}^{(k)}(t)$ and $i_{db}^{(k)}(t)$ are the outputs of the dynamic component k. The complete model is depicted in Figure 2.

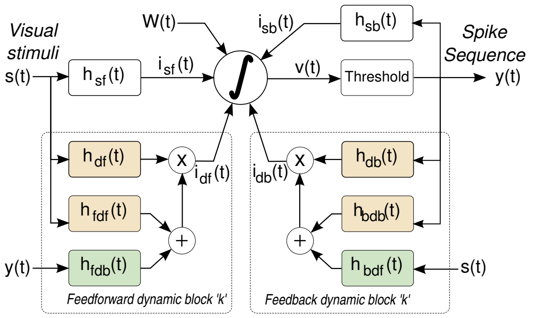


Figure 2: Dynamic model.

The outputs of both the static and the dynamic components are then added in the integrator block:

$$i(t) = i_{sf}(t) + i_{sb}(t) + \sum_k i_{df}^{(k)}(t) + \sum_k i_{db}^{(k)}(t) \tag{10}$$

and the subthreshold potential, $v(t)$, follows (2).

3.1 Model Implementation

To implement the model and estimate its parameters, it was discretised in time steps of $T_s = 1$ ms, leading to a discrete representation $x_n \equiv x[n]$ of the continuous signal $x(t)$. Basis functions were used for describing each of the filters in the model. While one can simply estimate the value of the filter for each sample nT_s, the use of basis functions allows: *i)* to reduce the dimensionality of the problem, thus accelerating convergence; and *ii)* to decrease overfitting of the parameters to the data used during the training step.

Each of the filters in the static component $h_x(t)$, $x \in \{sf, sb\}$, or in its discretised form, $\mathbf{h}_x = \left[h_x[1], \cdots, h_x[M]\right]$, where M is the filter memory, is defined by means of the basis functions $\mathfrak{B} = [\mathfrak{B}_1^\mathsf{T}, \cdots, \mathfrak{B}_B^\mathsf{T}]^\mathsf{T}$, $\mathfrak{B}_k = [b_{k1}, \cdots, b_{kM}]$ as:

$$\mathbf{h}_x = \sum_{m=1}^{B} a_{xm} \mathfrak{B}_m = \mathbf{A}_x^\mathsf{T} \mathfrak{B} \tag{11}$$

where $\mathbf{A}_x = [a_{x1}, \cdots, a_{xB}]^T$. In the case of the filters in the dynamic components, $h_x^{(1)}(t), \cdots, h_x^{(C)}(t)$, $x \in \{df, db, fdf, fdb, bdf, bdb\}$, one can organise them in a matrix $\mathbf{H}_x = \left[\mathbf{h}_x^{(1)\mathsf{T}}, \cdots, \mathbf{h}_x^{(C)\mathsf{T}}\right]^{\mathsf{T}}$, $\mathbf{h}_x^{(k)} = \left[h_x^{(k)}[1], \cdots, h_x^{(k)}[M]\right]$, where the filters are organised in rows and the samples in columns. The filtering matrix $\mathbf{H}_x$ can be computed as:

$$\mathbf{H}_x = \mathbf{A}_x^{\mathsf{T}} \mathfrak{B} \qquad (12)$$

For implementing the integrator linear filter, an infinite impulse response (IIR) description is used:

$$H_{IF}(z) = \frac{1-\beta}{1-\beta z^{-1}} \qquad (13)$$

where β defines a pole in $s_{\text{pole}} = \frac{1}{T}\log\beta$ and $\log(x)$ is the natural logarithm of x. For the system to remain stable, $\beta \in \Re_{]0,1[}$, such that $s_{\text{pole}} < 0$.

Notice that unlike in the continuous version of the model, the filter is now normalised with unitary DC gain. Thus, the variance of the noise, σ_n^2, at instant n, given that the last spike event was fired at instant n_0, becomes:

$$\sigma_n^2 = \beta^{2(n-n_0)}\sigma_0^2 + (1-\beta)^2\sigma^2 \sum_{k=0}^{n-n_0-1} \beta^{2k} = \\ = \beta^{2(n-n_0)}\sigma_0^2 + (1-\beta)^2 \frac{1-\beta^{2(n-n_0)}}{1-\beta^2}\sigma^2 \qquad (14)$$

3.2 Model Tuning

The complete set of parameters for the dynamic model in Figure 2 is:

$$\Big\{\mathbf{A}_{sf}, \mathbf{A}_{sb}, \mathbf{A}_{df}, \mathbf{A}_{db}, \mathbf{A}_{fdf}, \mathbf{A}_{fdb}, \mathbf{A}_{bdf}, \mathbf{A}_{bdb}, \\ \beta, \sigma, \sigma_0, V_{th}, V_0\Big\} \qquad (15)$$

However, some parameters depend on others. Namely, V_{th} and V_0 depend on the general gains of the model, $\mathbf{A}_x$. Changing their values implies changing the filter gains such that the total integration time for firing a spike remains the same. Also, modifying the value of β is similar to changing the shape of the other linear filters. Moreover, from our experience the initial noise variance σ_0^2 (the variance of the noise after the refractionary period) does not significatively influence the model tuning. Thus this parameter was removed from the learnable parameter set as well. The complete set of trainable parameters is therefore:

$$\Big\{\mathbf{A}_{sf}, \mathbf{A}_{sb}, \mathbf{A}_{df}, \mathbf{A}_{db}, \mathbf{A}_{fdf}, \mathbf{A}_{fdb}, \mathbf{A}_{bdf}, \mathbf{A}_{bdb}, \sigma\Big\} \qquad (16)$$

which corresponds to the basis functions coefficients for each filter and the noise standard deviation. The non-trainable parameters were set to: $\beta = 0.9$, $\sigma_0 = 0$, $V_{th} = 1$ and $V_0 = 0$.

For the optimisation of the model parameters, a Bayesian approach was applied to compute the probability of the spike sequence (Tomás and Sousa, 2007). Afterwards, gradient ascent was applied to maximise this probability. Following (15) from (Tomás and Sousa, 2007), the non null gradients in order to the parameters in (16) are:

$$\frac{du_n}{d\mathbf{A}_{sf}} = \mathbf{s}_n * \mathfrak{B} * h_{IF} \qquad (17a)$$

$$\frac{du_n}{d\mathbf{A}_{sb}} = \mathbf{y}_{n-1} * \mathfrak{B} * h_{IF} \qquad (17b)$$

$$\frac{du_n}{d\mathbf{A}_{df}} = \left[(\mathbf{s}_n * \mathfrak{B})(\mathbf{c}_{df})_n^{\mathsf{T}}\right] * h_{IF} \qquad (17c)$$

$$\frac{du_n}{d\mathbf{A}_{db}} = \left[(\mathbf{y}_{n-1} * \mathfrak{B})(\mathbf{c}_{db})_n^{\mathsf{T}}\right] * h_{IF} \qquad (17d)$$

$$\frac{du_n}{d\mathbf{A}_{fdf}} = \left[(\mathbf{s}_n * \mathfrak{B})(\mathbf{s}_n * \mathbf{A}_{df}^{\mathsf{T}}\mathfrak{B})^{\mathsf{T}}\right] * h_{IF} \qquad (17e)$$

$$\frac{du_n}{d\mathbf{A}_{fdb}} = \left[(\mathbf{y}_{n-1} * \mathfrak{B})(\mathbf{s}_n * \mathbf{A}_{df}^{\mathsf{T}}\mathfrak{B})^{\mathsf{T}}\right] * h_{IF} \qquad (17f)$$

$$\frac{du_n}{d\mathbf{A}_{bdf}} = \left[(\mathbf{s}_n * \mathfrak{B})(\mathbf{s}_n * \mathbf{A}_{df}^{\mathsf{T}}\mathfrak{B})^{\mathsf{T}}\right] * h_{IF} \qquad (17g)$$

$$\frac{du_n}{d\mathbf{A}_{bdb}} = \left[(\mathbf{y}_{n-1} * \mathfrak{B})(\mathbf{y}_{n-1} * \mathbf{A}_{df}^{\mathsf{T}}\mathfrak{B})^{\mathsf{T}}\right] * h_{IF} \qquad (17h)$$

$$\frac{1}{\sigma_n}\frac{d\sigma_n}{d\sigma} = 2(1-\beta)^2 \frac{1-\beta^{2(n-n_0)}}{1-\beta^2}\sigma \qquad (17i)$$

where $(\mathbf{c}_x)_n = [c_x^{(1)}[n], \cdots, c_x^{(C)}[n]]^{\mathsf{T}}$; the convolution, represented by the symbol $*$, operates along the columns of the operands; and the convolution with h_{IF} is performed using the IIR filter in (13), resetting the integration output to V_0 whenever a spike is fired.

However, unlike in static models, such as those described in (Tomás and Sousa, 2007), carefull initialisation is required in dynamic models. A method for doing this is to apply spike triggered analysis (Schwartz et al., 2002; Simoncelli et al., 2004). Eventhough this algorithm is valid under true Poisson neuron model, under IF models it becomes biased (Pillow and Simoncelli, 2003). This is mostly due to the IF natural non-linearities. However, since the estimation process of IF models has local minimums, it provides a good starting point.

Since the model includes feedforward and feedback dynamic mechanisms, Spike Triggered Average (STA) and Spike Triggered Covariance (STC) analysis were performed using both the stimuli and the spike history. The static filters h_{sf} and h_{sb} in Figure 2 were initially set with the shape of the feedforward and feedback STA, respectively. To adjust the initial

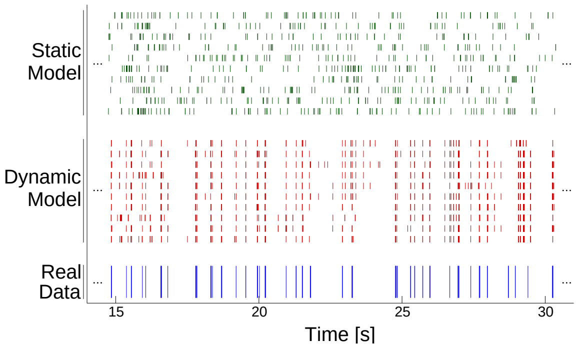


Figure 3: Spike response of the dynamic and static model *vs.* the response of real data (vertical lines represent the timing of the elicited spikes).

shape of the dynamic filters $h^{(k)}_{df}$ and $h^{(k)}_{db}$, excitatory and inhibitory directions can be extracted using STC analysis.

Table 1: Error measures between trained models and real responses.

		Spike Time	Inter Spike	NMSE
Static Model				
Training trial	mean	420.54	317.69	0.882
	std	10.39	7.05	
All four trials	mean	444.02	337.49	0.903
	std	12.03	6.95	
Dynamic Model				
Training trial	mean	251.63	242.62	0.625
	std	22.13	21.71	
All four trials	mean	325.85	281.32	0.808
	std	20.55	16.68	

mean - mean result std - standard deviation

4 EXPERIMENTAL RESULTS

The proposed training algorithm was implemented and tested to estimate the responses of real rabbit retina ganglion cells. The data set consists of four trials of full field white noise stimulus, where each trial has a duration of ≈ 50s with an average count of 6.58 spikes per second. While the stimuli values for these four trials is the same, small differences exist in the stimulation time: on average stimulation changed with a new random value at every 152 ms; the standard deviation of stimuli change was 238 ms. The visual stimuli was normalised by subtracting its mean value and then by dividing it by the standard deviation of itself. The resulting stimuli, which corresponds to the input s_n given to the model, is therefore a sequence of normally distributed random values with zero mean and unitary standard deviation.

To estimate the dynamic model, many basis functions can be used to describe the linear filters. Typical examples are the Laguerre bases (Akçay and Ninness, 1999; Tomás and Sousa, 2007) or sinusoidal basis (Keat et al., 2001). However, they are typically unable to describe delayed filters well and, for the used data, it considerably deteriorated the results. Thus, in the present work simple Gaussian kernels were used. While these bases are not orthogonal, they allow to significatively reduce the number of trainable parameters, whilst allowing to achieve good results. A total of 31 kernels were employed each separated by 10 ms, and having a standard deviation of 5.6 ms.

As depicted in subsection 3.2, for model tuning the static filters h_{sf} and h_{sb} were initially set with the shape of the STA applied to the stimuli and spike history, respectively. The dynamic filters h_{df} and h_{db} were set with the 5 most excitatory directions (extracted by using STC analysis); the experimental data showed no strong inhibitory directions. The modulating filters h_{fdf}, h_{fdb}, h_{bdf} and h_{bdb} were initially set to a small, non-zero value.

To compare the performance of the proposed dynamic model, we also performed fitting with a static model (the number of dynamic blocks in Figure 2 was set to zero). The training procedure was the same for both the static and the dynamic model. One of the ganglion cells' response trials was used for training. The other three were used for comparison. Again, stimulation times are not exactly the same for all trials, which leads to slightly different neuron responses.

For evaluating the performance of the models, 30 spike response trials were produced by using both the trained static and dynamic models. In Figure 3, we present the first 10 response trials obtained using the training data set. The figure also presents the real retina ganglion cells' response. Analysing the responses one can clearly see that the static model is unable to acquire the structure of the ganglion cells' response. On the other hand, the dynamic model is able to accurately reproduce the spike response pattern. However it does tend to fire 15% more spikes than the ganglion cell (average on the four trials).

For a better assessment of the response of the models, two error metrics proposed in (Victor and Purpura, 1997) were used. The first metric accounts for the cost associated with the absolute time of occurrence of neuronal events (Spike Time Metric). The second metric accounts for the cost of changing the intervals between two spikes (Inter Spike Metric). The movement cost q was set to 50 s^{-1} – see (Victor and Purpura, 1997). A firing rate metric based on the normalised mean squared error (NMSE) (Berry II and Meister, 1998) was also applied. For that purpose, the firing rates where estimated for both the real and the estimated data, by convolving their PeriStimulus Time Histogram (PSTH) (Berry II and Meister, 1998) with a gaussian window of zero mean and 20ms of standard deviation.

Cross-evaluation between the models' responses and the real ganglion cells responses using the described error metrics are presented in Table 1. This table shows the results when comparing the training trial and the other trials. The presented error metrics confirm our analysis: the dynamic model is able to capture the dynamics of real retina ganglion cells which cannot be described by a simple set of a feed-forward filter and a feedback filter. This can be seen by noticing that the mean values for all error metrics are much lower for the dynamic model than for the static model. The dynamic model also tends to achieve a higher degree of variability than the static model. However, this is due to the natural variability of the real data.

In the presented work, the number of dynamic components was fixed to five. However, for obtaining a more general model, one could start by using a larger number of dynamic components and then using feature selection – such as in (Tomás and Sousa, 2007) – to remove all unnecessary components.

5 CONCLUSIONS

Many researchers tend to classify SLIF models and Poisson-based models into different groups. However, as show in this paper, the former can be translated into the latest. One of the most important features of the SLIF model is the presence of feedback mechanisms. Given that this property helps to increase the spiking precision, we present a model whose coefficients are dynamic in time. Moreover we present a method to estimate its coefficients: it uses eigen-analysis to estimate the initial parameters and a gradient ascent technique for tuning the model. Presented results show that the proposed model is able to achieve better results than the simpler static models.

REFERENCES

Akçay, H. and Ninness, B. (1999). Orthonormal basis functions for modelling continuous-time systems. *Signal Processing*, 77(1):261–274.

Baccus, S. and Meister, M. (2002). Fast and Slow Contrast Adaptation in Retinal Circuitry. *Neuron*, 36(5):909–919.

Berry II, M. J. and Meister, M. (1998). Refractoriness and neural precision. *The Journal of Neuroscience*.

Bialek, W. and van Steveninck, R. (2005). Features and dimensions: Motion estimation in fly vision. *Arxiv preprint q-bio.NC/0505003*.

Capela, S., Tomás, P., and Sousa, L. (2007). Stochastic integrate-and-fire model for the retina. In *15th European Signal Processing Conference (EUSIPCO'2007)*.

Chichilnisky, E. J. (2001). A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12(2):199–213.

Gerstner, W., Jolivet, R., Brette, R., Clopath, C., Rauch, A., and Luscher, H. (2006). Predicting Neuronal Activity with Simple Models of the Threshold Type. *Time (msec)*, 100:50.

Keat, J., Reinagel, P., Reid, R. C., and Meister, M. (2001). Predicting every spike: A model for the responses of visual neurons. *Neuron*, 30:803–817.

Paninski, L. (2006). The most likely voltage path and large deviations approximations for integrate-and-fire neurons. *Journal of Computational Neuroscience*, 21(1):71–87.

Pillow, J. and Simoncelli, E. (2003). Biases in white noise analysis due to non-Poisson spike generation. *Neurocomputing*, 52(54):109–115.

Schwartz, O., Chichilnisky, E. J., and Simoncelli, E. P. (2002). Characterizing neural gain control using spike-triggered covariance. In Dietterich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems*, volume 14, pages 269–276, Cambridge, MA. MIT Press.

Simoncelli, E. P., Paninski, L., Pillow, J., and Schwartz, O. (2004). *Characterization of Neural Responses with Stochastic Stimuli*, chapter 23, pages 327–338. MIT Press.

Tomás, P. and Sousa, L. (2007). Feature selection for the stochastic integrate and fire model. In *2007 IEEE International Symposium on Intelligent Signal Processing (WISP'2007)*.

Victor, J. D. and Purpura, K. P. (1997). Metric-space analysis of spike trains: theory, algorithms and application. *Computation Neural Systems*, 8:127–164.