A Feature Selection Algorithm for the Regularization of Neuron Models

Pedro Tomás, *IEEE Student Member,* Leonel Sousa, *IEEE Senior Member,*

***Abstract*—This paper presents a novel training method for estimating the parameters of retina models, such as integrate-and-fire or Poisson-based. The presented models are constructed using a set of linear and nonlinear filters, described by basis functions and Taylor polynomials, respectively. This approach allows for the identification of a set of features that can be used for reproducing retina responses. By using the Bayesian-Laplace feature selection algorithm herein proposed, an efficient model with a reduced set of parameters is achieved. Experimental results show that the proposed algorithm is able to remove non-important features while still accurately reproducing retina responses. These results also show that the integrate-and-fire model is able to mimic the retina visual processing system using less parameters than the Poisson-based model.**

***Index Terms*—Biological system modeling, Point processes, Maximum likelihood estimation, Stochastic systems, Nonlinear systems, Nonlinear estimation**

I. Introduction

The modeling of the human retina has been a challenging research area and a topic of intense study in the last few years. The importance of this system is twofold. On the one hand, it allows researchers to study the connectivity of neural cells and to understand the network functionals. The advantage of this system, regarding the direct study of the human brain, is that it allows for an easy mapping between the input stimuli and the output response. On the other hand, the modeling of the human retina allows for the development of visual prostheses which could partially restore vision to blind people.

To tackle different types of blindnesses, visual prostheses with different characteristics have been proposed [1], namely: subretinal prostheses have been designed to replace degenerated photoreceptors while epiretinal prostheses replace higher functions of the retina; optic nerve prostheses allow for the restauration of vision when the retina is non-functional; visual cortex prostheses (e.g. [2]–[4]) are potentially capable of restoring vision to profoundly blind people, requiring only an healthy cortex, i.e. a cortex that, by means of self reorganization, is able to process the externally given stimulus [5].

The mammal retina is composed by several layers of neurons. The network connectivity of the cells in the different layers forms a visual processing path. At the output of the retina lie the ganglion cells that transmit visual information to the brain by means of a sequence of spikes (voltage pulses).

Many different approaches have been used for modeling the response of ganglion cells to visual stimuli, which typically adopt spike-triggered analysis [6], information-theoretic approaches [7] or maximum-likelihood estimation [8]. One of the problems of these approaches is that they tend to require a large number of parameters. In this sense, some attempts have been made to regularize neuron models, namely Poisson models estimated using spike-triggered analysis (STA) techniques [9,10]. However, to the best of our knowledge no general method has yet been proposed that can be applied to the different types of neuronal models. In this paper we propose to use of basis functions and Taylor expansions to provide an effective approach to describe the models' linear and nonlinear filters. Furthermore, we propose an algorithm to prune un-necessary model features, and show that the removal of these features from the model does not induce significant changes to the model's responses.

Neuron models can be organized in two main classes: Poisson-based models (e.g. [6,7,11]) and integrate-and-fire (IF) models (e.g. [8,12]). In [12] we focused on how to achieve a dimensionally reduced IF model, mainly because of two major advantages: *i)* if the final goal is to create an artificial retina (e.g. [13]), the achieved model is simpler to implement in hardware; and *ii)* a reduced parameter set model is less likely to become overfit.

In this paper we extend the feature selection method presented in [12] for Poisson based models. We show that the proposed feature selection algorithm is able to remove un-necessary features for both the Poisson and IF models without compromising their performance. The obtained results also allow us to conclude that the IF model achieves a better estimation of the retina visual processing system: it uses less parameters than the Poisson-based model.

This paper is organized as follows. Section II introduces the noisy leaky IF model and the Poisson-based model. Section III presents an algorithm for estimating the model parameters and for performing feature selection. The models and the proposed algorithm are tested by using real data from salamander retina in section IV. For assessing the performance of the training algorithm, we compare the responses of real ganglion cells with the responses of our models. A set of established error metrics are used to perform a quantitative analysis of the models' responses. Finally, section V concludes the paper.

II. Retina models

This section presents the two models used for evaluating the performance of the proposed training algorithm. The chosen

This paper is an extension of the one presented in the IEEE International Symposium on Intelligent Signal Processing (WISP'2007): *"Feature Selection for the Stochastic Integrate and Fire model"*. P. Tomás, and L. Sousa are with the Electrical and Computers Engineering Dept. at Instituto Superior Técnico, Technical University of Lisbon and with INESC-ID, Lisboa, Portugal. Address: Rua Alves Redol, 9, 1000-029 Lisboa, Portugal. Email: {pfzt,las}@inesc-id.pt; Phone: +351-213100300; FAX: +351-213145843. This work was supported by the Portuguese Foundation for Science and Technology (FCT), POSI/EEA-CPS/61779/2004.

models have systematically been used for mimicking the retina visual processing system (e.g. [6,14]–[16]); they have also been used for simulating the response of individual neurons (e.g. [7,8,14]).

The models are represented by their discrete counterparts because this allows for an easier implementation of the models in hardware. This is specially important for developing visual prosthesis. The models' response to visual stimuli is therefore given by a sequence of binary events $y[n]$ indicating weather a spike has been fired at time step n. Herein we state $y[n] = 1$ if a spike was fired at time step n, and $y[n] = 0$ otherwise.

A. Stochastic Leaky Integrate and Fire model

A typical approximation for neuron modeling is the Stochastic Leaky Integrate-and-Fire (SLIF) model. By adding a noise component to the model, it is possible to simulate the variability of real neurons [16]. The model has two periods of operation. During the integration period, it acts as a leaky integrator of the input current $i[n]$,

$$\begin{aligned} v[n] &= h_{IF}[n - n_{\text{last}}] * \Big(i[n] + W[n]\Big) \\ &= h_{IF}[n - n_{\text{last}}] * i[n] + \breve{W}[n] \end{aligned} \tag{1}$$

where n and n_{last} are, respectively, the time bins[1] of the current sample and of the sample where the last spike was elicited; h_{IF} is the impulsive response of a first order low pass filter; and $W[n] \sim N(0, \sigma)$ and $\breve{W}[n] = (W * h_{IF})[n]$ are normally distributed noise sources. When the potential $v[n]$ surpasses the value of a given threshold V_{th}, a spike is elicited by the model and the potential is reset – spiking period.

The probability for the SLIF model to fire a spike at time instant n is given by the probability for the noise to raise the subthreshold potential $v[n]$ above the threshold:

$$\begin{aligned} P_{spk}[n] &= P(\mathrm{E}\big[v[n]\big] + W[n] \geq V_{th}) \\ &= \mathrm{N}_{\text{cdf}}(\mathrm{E}\big[v[n]\big], V_{th}, \sigma_n) \end{aligned} \tag{2}$$

where $\mathrm{E}\big[v[n]\big] = h_{IF}[n - n_{\text{last}}] * i[n]$ is the expected value of $v[n]$ and N_{cdf} is the normal cumulative distribution function.

The input current $i[n]$ can be defined as a sum of three components (see Fig. 1): a feedback current $i_B[n]$ that depends on the spike history, a feedforward current $i_F[n]$ and a static input current μ. Although recent biological studies show that the first two components can be considered nonlinear functions with temporal dynamics, in this work we only apply a nonlinearity to the input stimuli. Experimental results presented in section IV show that this model is able to accurately reproduce the responses of the used ganglion cells.

The input component is therefore described by a two stage model: the input stimuli is transformed by means of a nonlinear function $m[n] = f(s[n])$, and the output is convolved with a linear filter with impulse response $h_F[n]$. The feedback component is described by a linear filter with impulse response $h_B[n]$. Finally, the spike response is obtained by using a threshold block that fires a spike whenever the potential $v[n]$ surpasses V_{th}. The above can be summarized by the following equations:

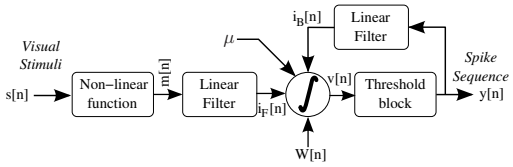

Fig. 1: Integrate and Fire model for the retina.

$$i[n] = \overbrace{h_F[n] * f(s[n])}^{i_F[n]} + \overbrace{h_B[n] * y[n-1]}^{i_B[n]} + \mu \tag{3a}$$

$$v[n] = \beta \cdot v[n-1] + (1-\beta) \cdot i[n] + \breve{W}[n] \tag{3b}$$

$$y[n] = H(v[n] - V_{th}) \tag{3c}$$

where β results from the conversion of the low pass filter h_{IF} in (1) to a first order infinite impulse response (IIR) filter with unitary gain. For model optimization, the nonlinear function is described by a Taylor series, while the filters $h_F[n]$ and $h_B[n]$ are constructed using orthogonal Laguerre basis functions [17] with z-domain transfer function:

$$H_L^{(k)}[z] = \frac{\sqrt{1-|\epsilon|}}{1-\epsilon z} \prod_{i=1}^{k-1} \frac{z-\epsilon}{1-\epsilon z} \quad \text{, for } k = 1, \cdots, M \text{ ,} \tag{4}$$

where ϵ regulates the position of the basis poles and zeros. Thus,

$$h_F[n] = \sum_{i=1}^{L} a_i h_L^{(i)}[n] = \mathbf{a}_F^{\mathsf{T}} \mathbf{h}_L[n] \tag{5a}$$

$$h_B[n] = \sum_{i=1}^{L} c_i h_L^{(i)}[n] = \mathbf{a}_B^{\mathsf{T}} \mathbf{h}_L[n] \tag{5b}$$

$$m[n] = \sum_{j=1}^{M} b_j s^j[n] = \mathbf{b}_f^{\mathsf{T}} \mathbf{x}[n] \tag{5c}$$

where $h_L^{(k)}[n]$ is the impulsive response of the filter with z-domain transfer function $H_L^{(k)}[z]$; $\mathbf{a}_F = [a_k]$, $\mathbf{b}_f = [b_k]$, $\mathbf{a}_B = [c_k]$, $\mathbf{h}_L[n] = [h_L^{(k)}]$ and $\mathbf{x}[n] = [s^k[n]]$ are column vectors; and $\mathbf{u}^{\mathsf{T}}$ denotes the transpose of vector $\mathbf{u}$. In other words, the letter $\mathbf{a}$ represents the column vector of coefficients representing a linear filter, while letter $\mathbf{b}$ is the column vector of coefficients describing a nonlinearity; subscripts F and B stand for feedforward filter and feedback filter, respectively; subscript f stands for the input nonlinearity $f(\cdot)$.

Considering that the model is always affected by noise, both during the integration and firing periods, one can write $\breve{W}[n] \sim N(0, \sigma_\infty)$ – see (9) and (10) in [12] –, where:

$$\sigma_\infty = \sigma \frac{1-\beta}{\sqrt{1-\beta^2}} \ . \tag{6}$$

Accordingly to the previous description, the complete set of parameters for this model would be:

$$\boldsymbol{\Theta}_{\text{IF}} = \{\mathbf{a}_F, \mathbf{a}_B, \mathbf{b}_f, \mu, \beta, V_0, V_{th}, \sigma\} \ . \tag{7}$$

However, a carefull analysis of the system can help to eliminate some of the model parameters. The set $\{V_0, V_{th}\}$ is

[1]Here the discrete counterpart of the model is directly used; information on the equations for continuous mode can be found in [12].

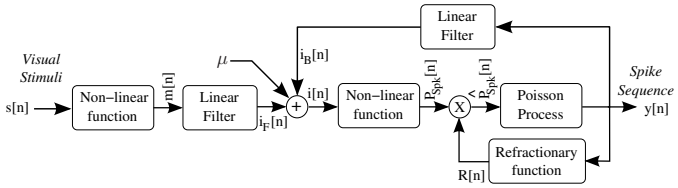

Fig. 2: Poisson-based model for the retina.

dependent on the gains $\mathbf{a}_F, \mathbf{a}_B, \mathbf{b}_f, \mu$; thus, in order to keep the same model response, increasing the difference $V_{th} - V_0$ implies increasing the gains of the system parameters; similarly, adding an offset to both V_{th} and V_0 implies a change in the constant input current μ. Thus we have set $V_0 = 0$ and $V_{th} = 1$.

The parameter β affects the behavior of the integration block in Fig. 1. However, a carefull examination of the model can help to exclude β from the set of tunable parameters. Since the convolution of two linear filters is a linear filter, one can simply fix the value for β and let the system temporal transfer function be defined by the filters h_F and h_B. Considering a sample time T, one can set β such that the module of the system pole (defined by $s_{\text{pole}} = \frac{1}{T}\log\beta$) is sufficiently high; the influence of known undesirable frequencies is reduced while the influence of unknown frequencies is defined by the coefficients of $\mathbf{a}_F$ and $\mathbf{a}_B$. In the presented experiments it was set $\beta = 0.9$ for a sampling time $T = 0.001$ s. Therefore, the tunable parameter set is therefore reduced from $\mathbf{\Theta}_{\text{IF}}$ to $\mathbf{\Theta}^*_{\text{IF}}$:

$$\mathbf{\Theta}^*_{\text{IF}} = \{\mathbf{a}_F, \mathbf{a}_B, \mathbf{b}_f, \mu, \sigma\} \ . \tag{8}$$

B. Poisson-Based model

The Poisson-based model is constructed using the same excitatory current $i[n]$ used by the SLIF model. However, the spike generation mechanism is different. For the output spiking probability to be limited to the range $[0\ 1]$, a sigmoid function is used (see Fig. 2):

$$P_{spk}[n] = \frac{1}{1 + e^{-g(i[n])}} \,, \tag{9}$$

where $g[n]$ is a nonlinear function described by Taylor polynomials

$$g(i[n]) = \sum_{k=1}^{K} d_k i^k[n] = \mathbf{b}_g^{\mathsf{T}}\mathbf{j}[n] \,, \tag{10}$$

and $\mathbf{b}_g = [d_k]$ and $\mathbf{j}[n] = [i^k[n]]$ are column vectors. Once again we use the letter $\mathbf{b}$ for describing a nonlinear function coefficients, where the subscript states the described function.

To add a refractory period to the neuron model, a characteristic of real neurons [14], a sigmoid-controlled feedback mechanism is introduced (see Fig. 2):

$$\breve{P}_{spk}[n] \ = P_{spk}[n]R[n - n_{\text{last}}] \tag{11a}$$

$$R[n] \ = \frac{1}{1+e^{-(n-n_0)T/\tau}} \tag{11b}$$

where n_{last} is the time instant of the last spike, T is the sampling period, and n_0 and τ regulate the refractory period.

Accordingly to the previous description, the complete set of parameters for this model is

$$\mathbf{\Theta}_{\text{P}} = \{\mathbf{a}_F, \mathbf{a}_B, \mathbf{b}_f, \mathbf{b}_g, \mu, n_0, \tau\} \,, \tag{12}$$

where $\mathbf{a}_F$ and $\mathbf{a}_B$ represent, as before, the vectors with the coefficients of the feedforward and feedback linear filters, respectively.

Notice that the discretized Poisson-based model is in fact a binomial process [14]. It can also be proved that the described SLIF model is also a special case of a binomial process [18]. Thus the difference between the two models is defined by how each model mimics the refractory period of neurons.

III. Model Optimization

Following a Bayesian approach, one can define the probability of the output $y_1, \cdots, y_n = \mathbf{y}_n$, $y_n \equiv y[n]$, as

$$P(\mathbf{y}_n|\mathbf{s}_n) = \prod_{n=1}^{M} P(y_n|\mathbf{y}_{n-1}, \mathbf{s}_n) \,, \tag{13}$$

where $\mathbf{s}_n = s_1, \cdots, s_n$, $s_n \equiv s[n]$, represents the visual stimuli. Also $P(y_n|\cdots)$ represents the probability for the estimated output $\hat{y}_n$ to be equal to the real output y_n. Once again, we assume $y_n = 1$ if a spike has been fired at time step n and $y_n = 0$ otherwise.

For model tuning, the parameter set $\mathbf{\Theta}$ that maximises the probability of the sequence must be chosen (or equivalently, the log-probability). Thus, differentiating the log-probability $\log P(\mathbf{y}_n|\mathbf{s}_n)$ regarding each of parameter, results:

$$\nabla l(\mathbf{\Theta}) = \sum_{n=1}^{M} \frac{\nabla P(y_n|\mathbf{y}_{n-1}, \mathbf{s}_n)}{P(y_n|\mathbf{y}_{n-1}, \mathbf{s}_n)} \,. \tag{14}$$

The gradient $\nabla P(y_n|\mathbf{y}_{n-1}, \mathbf{s}_n)$ is taken by computing the partial derivatives of the spiking probability in (2) or (11), for the SLIF or Poisson-based model, respectively. In the presented case, a gradient ascent procedure using adaptative steps [19] is used to fit the parameters for each of the models.

A. Applying Bayesian feature selection

So far we have described the models used for estimating the output of the retina ganglion cells. Additionally we have presented an optimization function for tuning the model parameters. However, we have neither discussed how many basis functions should one use to model the retina processing system, nor the shape of these functions (parameter ϵ in (4)). Since it is not easy to optimize ϵ, a better way for tuning the model is to generate a set of basis functions and then to select the minimum set of functions that best describe the ganglion cells' transfer function. While such a method has been an intense topic of research and has been applied in several areas, it has never been used for retina modeling.

Suppose that we have a set of candidate models $\mathcal{M}$ each described by set of parameters $\mathbf{\Theta}_m = \{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_{K_m}\}$ (not necessarily of equal size). Using a Bayesian procedure, the best model would be the one that maximises the joint probability of the output sequence $\mathbf{y}$ and the model m:

$$p(\mathbf{y}, m|\mathbf{s}) = p(m) \int_{\mathbf{\Theta}_m} p(\mathbf{y}|\mathbf{s}, \mathbf{\Theta}_m, m) p(\mathbf{\Theta}_m|m) d\mathbf{\Theta} \,. \tag{15}$$

Since the integral is not always easy to compute, one can use the large-scale Bayesian-Laplace approximation [20,21]:

$$L(\mathbf{\Theta}_m; m) = \log p(\mathbf{y}|\mathbf{s}, \mathbf{\Theta}_m, m) - \frac{d}{2}\log N\,, \qquad (16)$$

where d is the dimension of $\mathbf{\Theta}_m$ (number of non-null parameters) and N is the number of valid output samples, i.e., it is the number of samples for which there is knowledge on both the stimuli input and the spike history.

The above equation assumes that: *i)* there is no prior knowledge on the best model for our data, *ii)* that $p(\mathbf{\Theta}_m|m)$ is sufficiently flat around $\hat{\mathbf{\Theta}}$, and *iii)* the model parameters are independent. While the third condition is not satisfied by the presented models, the optimization function can still be used and the achieved results are good, as it will be shown in section IV.

B. Optimization algorithm

One of the problems in feature selection is the lack of a differentiable function on the model, or, in our specific case, on the number of features. A method to overcome this difficulty is to iteratively add/remove features from the current solution and then to maximise $p(\mathbf{y}|\mathbf{s})$ given the current set of parameters. Such a method is the base of Algorithm 1 and works as follows. First an initial model m^* is created using a large number of basis functions and a high order for the Taylor polynomials. To avoid numerical representation problems in the training of the SLIF model, the noise standard deviation σ must also be initialized such that $\sigma > \sqrt{1-\beta^2}/4$. In both models the coefficients are initially set to zero with the exception of the first term for the Taylor polynomials: $b_1 = 1$ and $d_1 = 1$ (if applicable).

During the first phase of Algorithm 1, the initial model m^* is tunned by using (14) to find the parameters $\mathbf{\Theta}^*$ that maximise (13). Adaptative steps [19] are used to speed up the convergence of the gradient ascent process. After the initial parameter estimation procedure, the algorithm iteratively tries to find new models $\hat{m}$ such that $L(\hat{\mathbf{\Theta}}, \hat{m}) > L(\mathbf{\Theta}^*, m^*)$. Each new model $\hat{m}$ is created from the *previous best* model m^* and then modified by increasing/decreasing either the number of basis functions or the order of the nonlinear function.

Let $\mathcal{N}(\mathbf{a}_F)$ and $\mathcal{N}(\mathbf{a}_B)$ represent the number of basis functions that model m uses to describe the filters h_F and h_B (i.e. the number of non-null coefficients in $\mathbf{a}_F$ and $\mathbf{a}_B$). Additionally, let $\mathcal{N}(\mathbf{b}_f)$ and $\mathcal{N}(\mathbf{b}_g)$ be the order of the nonlinearity used to describe $f(\cdot)$ and $g(\cdot)$. To search for new models $\hat{m}$, Algorithm 1 uses the following steps:

1) try to use one less/more basis function to represent h_F, $\hat{\mathcal{N}}(\mathbf{a}_F) = \mathcal{N}(\mathbf{a}_F) \mp 1$;
2) try to use one less/more basis function to represent h_B, $\hat{\mathcal{N}}(\mathbf{a}_B) = \mathcal{N}(\mathbf{a}_B) \mp 1$;
3) try to decrease/increase the order of the Taylor polynomial used to represent $f(\cdot)$, $\hat{\mathcal{N}}(\mathbf{b}_f) = \mathcal{N}(\mathbf{b}_f) \mp 1$;

(Only for the case of the Poisson-based model)

4) try to decrease/increase the order of the Taylor polynomial used to represent $g(\cdot)$, $\hat{\mathcal{N}}(\mathbf{b}_g) = \mathcal{N}(\mathbf{b}_g) \mp 1$.

Algorithm 1 Model optimization algorithm

< INITIALIZE MODEL m^* >
Generate the set of 20 basis functions;
Set initial nonlinearity order: $\mathcal{N}^*(\mathbf{b}_f) = \mathcal{N}^*(\mathbf{b}_f) = 10$
Set the number of basis functions used for filters h_F and h_B: $\mathcal{N}^*(\mathbf{a}_F) = \mathcal{N}^*(\mathbf{a}_B) = 20$
Initialize the parameters $\mathbf{\Theta}$ for initial model m^*:
$V_{th} \leftarrow 1$, $\beta \leftarrow 0.9$, $\mu \leftarrow 0$, $\sigma \leftarrow 0.2$ (*For the SLIF model*)
$\mu \leftarrow 0$, $n_0 \leftarrow 0$, $\tau \leftarrow 0.001$, $\forall_{k\neq 1} d_k \leftarrow 0$, $d_1 \leftarrow 1$ (*For the Poisson-based model*)
$\mathbf{a}_F \leftarrow \mathbf{0}$, $\mathbf{a}_B \leftarrow \mathbf{0}$, $\forall_{k\neq 1} b_k \leftarrow 0$, $b_1 \leftarrow 1$ (*For both models*)

< MODEL OPTIMIZATION >
$(\mathbf{\Theta}^*, m^*, \text{IsBetter}) \leftarrow \text{optim}(\mathbf{\Theta}^*, m^*, m^*, \text{false})$
repeat
  IsBetter $\leftarrow$ false
  STEP 1:
  $\hat{m} \leftarrow m^*$, $\mathcal{N}^*(\mathbf{a}_F) \leftarrow \hat{\mathcal{N}}(\mathbf{a}_F) \mp 1$
  $(\mathbf{\Theta}^*, m^*, \text{IsBetter}) \leftarrow \text{optim}(\mathbf{\Theta}^*, \hat{m}, m^*, \text{IsBetter})$
  STEP 2:
  $\hat{m} \leftarrow m^*$, $\mathcal{N}^*(\mathbf{a}_B) \leftarrow \hat{\mathcal{N}}(\mathbf{a}_B) \mp 1$
  $(\mathbf{\Theta}^*, m^*, \text{IsBetter}) \leftarrow \text{optim}(\mathbf{\Theta}^*, \hat{m}, m^*, \text{IsBetter})$
  STEP 3:
  $\hat{m} \leftarrow m^*$, $\mathcal{N}^*(\mathbf{b}_f) \leftarrow \hat{\mathcal{N}}(\mathbf{b}_f) \mp 1$
  $(\mathbf{\Theta}^*, m^*, \text{IsBetter}) \leftarrow \text{optim}(\mathbf{\Theta}^*, \hat{m}, m^*, \text{IsBetter})$
  STEP 4 *(for the Poisson-based model only)*:
  $\hat{m} \leftarrow m^*$, $\mathcal{N}^*(\mathbf{b}_g) \leftarrow \hat{\mathcal{N}}(\mathbf{b}_g) \mp 1$
  $(\mathbf{\Theta}^*, m^*, \text{IsBetter}) \leftarrow \text{optim}(\mathbf{\Theta}^*, \hat{m}, m^*, \text{IsBetter})$
until IsBetter = false

< AUXILIARY OPTIMIZATION FUNCTION >
function $(\mathbf{\Theta}^*, m^*, \text{IsBetter}) = \text{optim}(\mathbf{\Theta}^*, m, m^*, \text{IsBetter})$
  Set: $\hat{\mathbf{\Theta}} \leftarrow \arg_{\mathbf{\Theta}} \max \log\ p(\mathbf{y}|\mathbf{s}, \mathbf{\Theta}, m)$
  Set: $\hat{L} \leftarrow \log\ p(\mathbf{y}|\mathbf{s}, \hat{\mathbf{\Theta}}, m)$
  if $L(\hat{\mathbf{\Theta}}, m) > L(\mathbf{\Theta}^*, m^*)$ **then**
    Set: $\mathbf{\Theta}^* \leftarrow \hat{\mathbf{\Theta}}$
    Set: $m^* \leftarrow m$
    Set: IsBetter $\leftarrow$ true
  end if
end function

Notice that in the above algorithm using one less basis function is equivalent to forcing the parameter for that basis function to be zero.

In Algorithm 1, for each step 1-4, once a new model has been found such that $L(\hat{\mathbf{\Theta}}, \hat{m}) > L(\mathbf{\Theta}^*, m^*)$, the algorithm proceeds to the next step. Thus, for example, after removing one forward basis function it only tries to remove another after going through steps 2-4. Moreover, when removing one basis function (steps 1 and 2) one should first try to remove the basis having the least influence on the model. To assess basis

influence, the algorithm computes their total power ψ:

$$\begin{cases} \psi(a_k) = |a_k| \sqrt{T \sum_n \left(h_L^{(k)}[n] \right)^2} & \text{, for feedforward basis} \\ \psi(c_k) = |c_k| \sqrt{T \sum_n \left(h_L^{(k)}[n] \right)^2} & \text{, for feedback basis.} \end{cases} \tag{17}$$

If the removal of this basis function does not lead to a decrease in the optimization function, the algorithm tries to remove the basis function that has the second lowest power; the algorithm continues to iterate until a basis function has been removed or until no basis function is left to be removed. For the experimental data set used in this paper, the use of (17) allows to significatively decrease the convergence time: if any basis function can be removed, the algorithm is able to find it in the first iterations.

IV. Experimental Results

The proposed optimization algorithm was implemented and tested in MATLAB with the experimental data used in [16]. These data consists of 12 trials of full field white noise stimulation for a salamander ON retinal ganglion cell (RGC), where each trial has a duration of 10 seconds with an average count of 8.34 spikes per second. The visual stimuli was normalized by subtracting its mean value and then diving by its standard deviation. The resulting stimuli, which corresponds to the model input $s[n]$, is therefore a sequence of normally distributed random values with zero mean and unitary standard deviation. When presenting the experimental results, these data will be designated as RGC.

These data was used for tuning the parameters of several SLIF and Poisson-based models. The differences between the experimented models are related with the selected basis functions and the leaky integrator coefficient β. The values for the experimented parameters ranged in the intervals $0.6 \leq \epsilon \leq 0.9$ and $0.7 \leq \beta \leq 0.95$. Experimental tests ranged from using orthogonal and non-orthogonal basis (by using basis extracted with different values of ϵ). Our tests revealed nonsignificant differences in the models' responses when varying the parameters ϵ and β.

In order to evaluate the performance of the feature selection algorithm and compare the spiking mechanisms, results are presented for one SLIF model and one Poisson-based model. A set of 20 basis functions are used for both the feedforwad and feedback filters while the order of the nonlinearities was initially set to 10 – as shown in Algorithm 1. The initial SLIF and Poisson models are henceforth referred as F-IF and F-P, respectively. After applying the proposed feature selection mechanism, the number of parameters of the SLIF model, henceforth designated as R-IF, was reduced from an initial value of 52 to only 18. The dimensions of the reduced model are: $\mathcal{N}(\mathbf{a}_F) = 10$, $\mathcal{N}(\mathbf{a}_B) = 5$, $\mathcal{N}(\mathbf{b}_f) = 1$. The number of parameters was also reduced in the case of the Poisson-based model from an initial value of 63 to a value of 23, resulting in a reduced model with $\mathcal{N}(\mathbf{a}_F) = 8$, $\mathcal{N}(\mathbf{a}_B) = 7$, $\mathcal{N}(\mathbf{b}_f) = 4$, $\mathcal{N}(\mathbf{b}_g) = 1$. This model will be referred as R-P. These results are presented in Table I. The other models in this table, F-P2 and R-P2, will be used for comparison later on.

TABLE I: Number of features per model.

Model name	Spiking mechanism	Model type	$\mathcal{N}(\mathbf{a}_F)$	$\mathcal{N}(\mathbf{a}_B)$	$\mathcal{N}(\mathbf{b}_f)$	$\mathcal{N}(\mathbf{b}_g)$
F-IF	SLIF[c]	Full[a]	20	20	10	–
R-IF		Reduced[b]	10	5	1	–
F-P	Poisson[d]	Full[a]	20	20	10	10
R-P		Reduced[b]	8	7	4	1
F-P2	Poisson[d]	Full[a]	20	20	1	10
R-P2		Reduced[b]	11	8	1	1

[a] Initial model, i.e. before applying Algorithm 1
[b] Final model, i.e. after applying Algorithm 1
[c] See Figure 1
[d] See Figure 2
– Not applicable

After tuning the model, 100 spike sequence trials were generated for each of the two types of models. The first 12 trials for each model are presented in Fig. 3. The real responses from the retina ganglion cell are also presented in this figure; vertical lines represent the time instants were spikes were generated while the shaded area represents the mean firing rate for the 100 trials. These rates were estimated by convolving their PeriStimulus Time Histogram (PSTH) [6] with a Gaussian window of zero mean and 25ms of standard deviation.

Analyzing the spike trains and the mean firing rates in Fig. 3 one can qualitatively conclude that all models are able to simulate the behavior of the salamander retina. Comparing the firing rates of each model with the real data, i.e. the shaded areas in Fig. 3, one can conclude that almost all spiking events of the real RGC are well simulated by any model. However, the full featured models (F-IF and F-P) tend to create spiking events where none exists, e.g., around 3.8s. In some cases, theses events are removed after applying feature selection. Nonetheless, in general the mean squared error of the mean firing rate (between the RGC and the models) slightly increases when applying feature selection – this will be illustrated by applying the normalized mean squared error (NMSE) to quantitatively assess the results that will be presented later on. However, all models have similar behaviors, showing that the optimization algorithm is really able to remove unnecessary model parameters without compromising its response. This will be further validated by using neural metrics.

To perform an analytical evaluation of the models, their responses to visual stimuli was compared against real data using two spike train metrics proposed in [22]. One of these metrics accounts for the cost associated with the absolute time of occurrence of neuronal events (Spike Time Metric). The other metric accounts for the cost of changing the intervals between two spikes (Inter Spike Metric). The cost of moving a spike was set to $q = 50\ s^{-1}$ (see [22]). The NMSE [15], a firing rate metric, was also used as an auxiliary distance measure: it measures the difference between the models by computing the mean squared error of the firing rates in Fig. 3 and by dividing this result by the variance over time of the RGC firing rate. The performance evaluation was made using 12 trials from the real data set and 100 trials from our models. Values were obtained for inner evaluation of responses, i.e. when the responses of the models/cell are compared with themselves – see Table II –, and when performing cross evaluation of models responses,

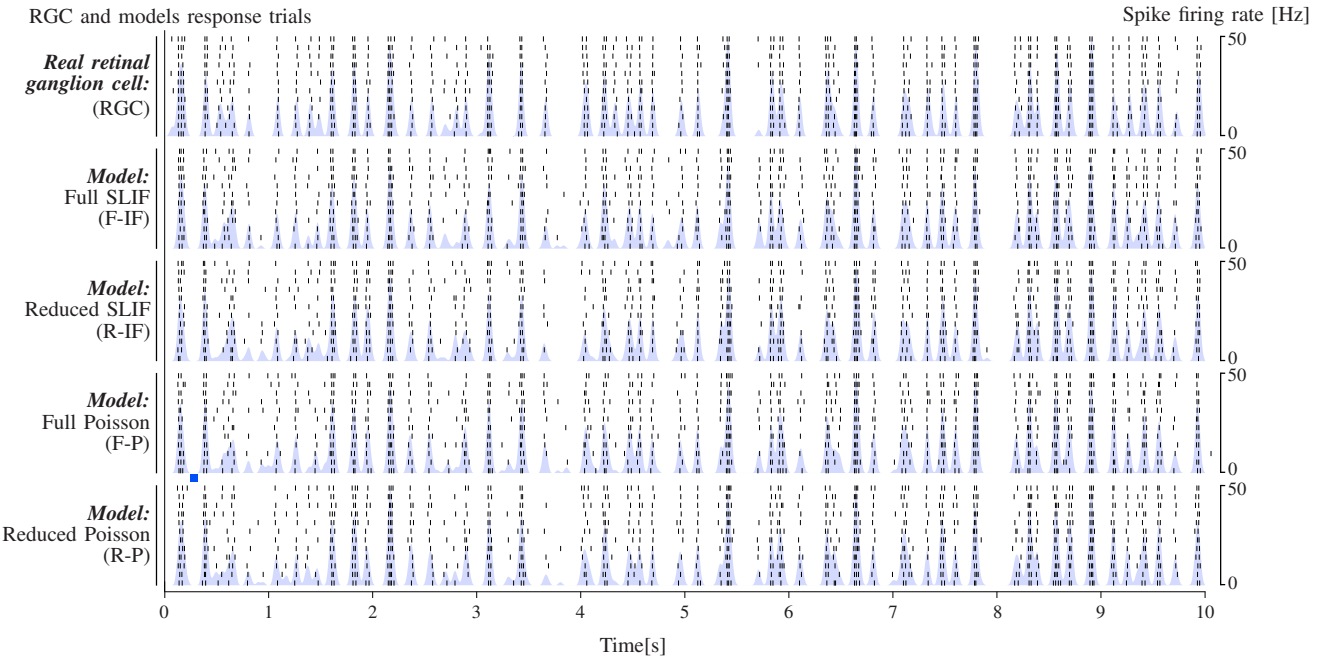

Fig. 3: Twelve response trials for both real retinal ganglion cell and the presented models; each vertical line represents the time step where a spike was elicited; each rows of vertical lines (spikes) represents a different trial; the shaded areas represent the mean firing rate for each model.

i.e. by comparing the responses of the models against those from the real retinal ganglion cell – see Table III.

From the values presented in Tables II and III, it can be concluded that the proposed algorithm for tuning the parameters of the retina models is able to reduce the number of model parameters without significatively affecting the model response. In [12] it has already been shown that the proposed feature selection algorithm could be applied with the SLIF model presented in section II. Here it is shown that it can also be used with other types of models such as the Poisson-based model.

The obtained results also show that the leaky integrator is a better estimator for the neurons refractory period than the logistic function. While the results are identical in terms of the used error measures, the SLIF model leads to a smaller number of parameters. It can also be concluded that when the proposed feature selection algorithm is applied to the Poisson-based model, we were unable to decrease the order of the input nonlinearity to one (making $f(\cdot)$ a linear function).

To compare the results when the input function $f(\cdot)$ of the Poisson-based model is linear, $f(s[n]) = s[n]$, we reapplied the algorithm to a model using $\mathcal{N}(\mathbf{b}_f) = 1$ (represented as F-P2 in Table I). Using this new F-P2 model as a starting point for Algorithm 1, we obtained a new regularized model, the R-P2. The results are also presented in Table II and Table III. As it can be seen, while the error measures are mainly the same, the proposed algorithm required more basis functions to successfully describe the response of the salamander retina.

It is interesting to realize that the overall number of parameters is the same for both Poisson-based models, the R-P and the R-P2. This means that, by removing the input nonlinearity, the proposed algorithm requires more basis functions to achieve a similar solution. Once again, this shows the effectiveness of the algorithm for regularizing the neuron models: it gives a solution using the least number of parameters. Notice that the existence of multiple solutions has to do with the fact that the parameters for the presented neuron models are not independent. This is due to the nonlinearity inherent to the spiking mechanism of the neuron models.

TABLE II: Inner evaluation of cell/model responses.

Data Source	Number of Spikes[(a)]	Spike Time[(a)]	Inter Spike[(a)]
RGC	83.42 ± 2.81	42.19 ± 3.58	59.05 ± 5.26
F-IF	83.04 ± 2.72	56.88 ± 4.43	71.98 ± 4.84
R-IF	83.31 ± 2.90	57.99 ± 4.44	72.93 ± 5.16
F-P	81.86 ± 2.32	57.35 ± 4.68	72.56 ± 5.21
R-P	82.58 ± 3.06	58.51 ± 4.26	74.59 ± 4.80
F-P2	82.19 ± 2.89	58.21 ± 4.76	73.06 ± 4.89
R-P2	82.66 ± 2.81	58.34 ± 4.73	73.49 ± 5.03

[(a)] The values are presented in the format: *mean* ± *standard deviation.*

TABLE III: Cross evaluation of model responses – comparing with RGC data.

Data Source	Spike Time[(a)]	Inter Spike[(a)]	NMSE[(b)]
F-IF	58.31 ± 3.70	74.59 ± 5.07	0.080
R-IF	57.71 ± 4.38	73.12 ± 4.96	0.092
F-P	57.28 ± 4.32	73.10 ± 5.16	0.086
R-P	58.28 ± 4.23	74.59 ± 5.26	0.088
F-P2	58.16 ± 4.21	73.40 ± 5.00	0.092
R-P2	57.81 ± 4.41	73.93 ± 5.24	0.088

[(a)] The values are presented in the format: *mean* ± *standard deviation.*
[(b)] Since all trials are evaluated simultaneously, it is not possible to retreive mean and standard deviation.

V. Conclusions

This paper presents a feature selection algorithm for automatically reducing the dimensionality of retina models. To test the proposed algorithm, a Poisson-based retina model and a Stochastic Leaky Integrate-and-Fire (SLIF) retina model were used. By applying the proposed algorithm to tune the model parameters we were able to reduce the dimensionality of both models. The obtained results using different biological plausible error measures allow us to conclude that the proposed algorithm is able to eliminate model features without changing the models' responses. This is an important step for mimicking a neuron's transfer function and for creating a prosthesis to circumvent a human impairment: it allows to remove features which are not original in the neuron, but are artificially introduced by the general models.

Experimental results also show that the SLIF model is able to describe the response of the salamander retina using a lower number of parameters than the Poisson-based model. This comes from the fact that the leaky integrator with reset used by the SLIF model is a better estimator for the neuron refractory period than the sigmoid function used by the Poisson-based model.

VI. Acknowledgments

The experimental data used in this paper was gently provided by Prof. Markus Meister.

References

[1] R. R. Lakhanpal, D. Yanai, J. D. Weiland, G. Y. Fujii, S. Caffey, R. J. Greenberg, E. J. de Juan, and M. S. Humayun, "Advances in the development of visual prostheses," *Advances in the development of visual prostheses*, vol. 14, no. 3, pp. 122–127, June 2003.

[2] W. H. Dobelle, "Artificial vision/human implantation program," in *48th Annual Conference of the American Society for Artificial Internal Organs*, New York, June 2002.

[3] V. Chowdhury, J. Morley, and M. Coroneo, "Surface stimulation of the brain with a prototype array for a visual cortex prosthesis," *J Clin Neurosci*, vol. 11, no. 7, pp. 750–755, September 2004.

[4] M. Piedade, J. Gerald, L. Sousa, G. Tavares, and P. Tomas, "Visual neuroprosthesis: A non-invasive system for stimulating the cortex," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 52, pp. 2648–2662, December 2005.

[5] E. Fernandez, F. Pelayo, S. Romero, M. Bongard, C. Marin, A. Alfaro, and L. Merabet, "Development of a cortical visual neuroprosthesis for the blind: the relevance of neuroplasticity," *J Neural Eng*, vol. 2, no. 2, pp. R1–R12, 2005.

[6] E. J. Chichilnisky, "A simple white noise analysis of neuronal light responses," *Network: Computation in Neural Systems*, vol. 12, no. 2, pp. 199–213, May 2001.

[7] J. W. Pillow and E. P. Simoncelli, "Dimensionality reduction in neural models: an information-theoretic generalization of spike-triggered average and covariance analysis," *Journal of Vision*, vol. 6, no. 4, pp. 414–428, April 2006.

[8] L. Paninski, "The most likely voltage path and large deviations approximations for integrate-and-fire neurons," *Journal of Computational Neuroscience*, vol. 21, no. 1, pp. 71–87, April 2006.

[9] J. Pillow and E. Simoncelli, "Dimensionality reduction in neural models: An information-theoretic generalization of spike-triggered average and covariance analysis," *Journal of Vision*, vol. 6, no. 4, pp. 414–428, 2006.

[10] T. Sharpee, "Comparison of objective functions for estimating linear-nonlinear models," *Arxiv preprint arXiv:0801.0311*, 2008.

[11] S. Martins, L. Sousa, and J. C. Martins, "Additive logistic regression applied to retina modelling," in *IEEE International Conference on Image Processing (ICIP)*, September 2007, pp. III–309–III–312.

[12] P. Tomás and L. Sousa, "Feature selection for the stochastic integrate and fire model," in *2007 IEEE International Symposium on Intelligent Signal Processing (WISP 2007)*. IEEE, October 2007, pp. 581–586.

[13] M. Piedade, J. Gerald, L. Sousa, G. Tavares, and P. Tomás, "Visual neuroprosthesis: A non-invasive system for stimulating the cortex," *IEEE Transactions on Circuits and Systems I - Regular papers*, vol. 6, no. 12, pp. 2648–2662, December 2005.

[14] W. Gerstner and W. Kistler, *Spiking Neuron Models: single neurons, populations, plasticity*. Cambridge University Press, 2002.

[15] M. J. Berry II and M. Meister, "Refractoriness and neural precision," *The Journal of Neuroscience*, pp. 2200–2211, 1998.

[16] J. Keat, P. Reinagel, R. C. Reid, and M. Meister, "Predicting every spike: A model for the responses of visual neurons," *Neuron*, vol. 30, pp. 803–817, June 2001.

[17] H. Akçay and B. Ninness, "Orthonormal basis functions for modelling continuous-time systems," *Signal Processing*, vol. 77, no. 1, pp. 261–274, 1999.

[18] P. Tomás, J. C. Martins, and L. Sousa, "Towards a unified model for the retina: Static vs dynamic integrate and fire models," in *2008 International Conference on Bio-Inspired Systems and Signal Processing (BIOSIGNALS 2008)*, January 2008, pp. 528–533.

[19] L. B. Almeida, T. Langlois, J. D. Amaral, and A. Plakhov, *Parameter Adaptation in Stochastic Optimization*. Cambridge University Press, 1998, ch. 6.

[20] J. Oliver and R. Baxter, "MML and Bayesianism: Similarities and differences (introduction to Minimum Encoding Inference - part II)," Department of Computer Science, Monash University, Tech. Rep. 206, 1994.

[21] A. D. Lanterman, "Schwarz, Wallace, and Rissanen: intertwining themes in theories of model selection," *International Statistical Review*, vol. 69, no. 2, pp. 185–212, 2001.

[22] J. D. Victor and K. P. Purpura, "Metric-space analysis of spike trains: theory, algorithms and application," *Computation Neural Systems*, vol. 8, pp. 127–164, 1997.

Pedro Tomás (S' 04, M' 09) received his diploma degree, MSc and PhD in electrical and computer engineering from the Instituto Superior Técnico (IST), Technical University of Lisbon, Portugal, in 2003, 2006 and 2009, respectively. Currently he is developing his research work with the Signal Processing Systems Group (SIPS) at Instituto de Engenharia de Sistemas e Computadores–R&D (INESC-ID). He is also a Visiting Professor at ISCTE (Instituto Superior das Ciências do Trabalho e da Empresa). His research activities include statistical signal processing, modeling of physiological systems and algorithms for estimation and classification. He is also interest in general signal processing and in computer architectures, parallel and distributed computing.

Leonel Augusto Sousa (M' 01-SM' 03) received the diploma degree in Electronic and Telecommunication Engineering from Universidade de Aveiro (Portugal) in 1984. He received the MSc and PhD degrees in Electrical and Computer Engineering from the Instituto Superior Técnico (IST), Technical University of Lisbon (Portugal), in 1989 and 1996, respectively. He is currently an associate professor of the Electrical and Computer Engineering Department at IST, and a Senior Researcher at Instituto de Engenharia de Sistemas e Computadores–R&D (INESC-ID). His research interests include computer architectures, parallel and distributed computing, and multimedia and biomedical systems. He has contributed more than 150 papers to international journals and conferences and he is a member of HiPEAC European Network of Excellence. He is a member of the IEEE Computer Society and a member of ACM.