Signal Coding and Reconstruction Using Deterministic Spiking Neurons

Gokhan Kaya
Department of Informatics
Gebze Technical University
Kocaeli, Turkey
Email: gokayago@gmail.com

Arunava Banerjee
Computer & Information Science & Engineering
University of Florida
Gainesville, FL 32611
Email: arunava@cise.ufl.edu

***Abstract*—The problem of deterministically coding a continuous time signal using an ensemble of spike trains is addressed. Coding is defined with an eye toward "efficiency", defined as a trade-off between the number of spikes in the code and the quality of the code operationalized using the notion of reconstruction error. It is shown that inverting the coding model leads to a reconstruction procedure that amounts to a constrained optimization problem. A class of coding models is considered that makes the coding procedure biologically plausible while at the same time making the reconstruction problem tractable. It is demonstrated that the reconstruction error depends acutely on the coding model. This tight coupling is then used to describe a procedure that learns a coding model of improved efficiency. Experiments on a corpus of voice data validate the strength of our approach.**

I. Introduction

Brains of most animal species receive sensory stimuli in the form of spike trains. Take the human visual system, for example. The spatio-temporal light signal impinging on the photoreceptor cells of the retina, after several stages of processing through the horizontal, bipolar, and amacrine cells, is turned into spike trains at the retinal ganglion cells. The axons of the ganglion cells, that form the optic nerve then communicate these spike trains to the brain. Other sensory modalities have corresponding counterparts.

How a continuous time signal is transformed into spike trains, how much of the information in the continuous time signal is present in the spike trains, how the quality of this information depends on various aspects of the translation process, etc., are therefore some of the most fundamental questions in Computational Neuroscience.

These questions of signal coding in animal sensory systems have been the subject of intense effort in the past, albeit within a *stochastic* framework. For example, recent progress has been based on the paradigm of generalized linear models [3], [8], [5], a particularly popular subclass of which is the linear-nonlinear (LN) cascade. Here the sensory stimulus is turned into a stochastic spike train via an instantaneous firing rate model. The firing rate of the neuron is computed by first applying a linear causal filter to the input signal, followed by a static non-linearity. Spikes are then generated stochastically from the instantaneous firing rate. Whereas this stochastic viewpoint has seen wide applicability, there remain examples such as visual coding in the H1 neuron of the fly [4] where the transformation has been shown to be deterministic. In this paper, several questions revolving around signal coding using spike trains is formally addressed in a *deterministic* setting.

The "efficiency" of a coding mechanism depends entirely on the constraints faced by an organism in its natural environment. For an organism for which energy is a scarce resource, the number of spikes (or the spike rate) used to code a continuous time signal determines efficiency; the sparser the spike train, the more efficient the code. On the other hand, for animals whose survival depends on high sensory acuity, the efficiency of the code depends on how much information the spike train carries about the sensory signal. By taking a coding-reconstruction point of view, our approach gets around the question of efficiency by framing it as a trade-off between the number of spikes in the code and the quality of the code operationalized using the notion of reconstruction error.

Reconstruction is formally posed as inverting the coding model which leads to a novel (coding) *model-aware* framework. Analysis of the reconstruction demonstrates that the reconstruction error depends acutely on the coding model. This coupling then suggests a procedure that learns a coding model of improved efficiency. The framework proposed is therefore consistent with the view that animals "tune" their sensory systems to their natural environment.

The remainder of the paper is structured as follows. Section II formally describes the coding transformation. It presents a model of an artificial neuron and specifies how the neuron generates spikes given an input signal. Section III then describes how a spike train and the model of the neuron can be used to obtain a candidate reconstruction for the input signal. The importance of selecting a suitable kernel (instantiating the neuron model) is highlighted and, for a special scenario, the best kernel for optimal reconstruction is derived. Section IV then described the general coupled optimization problem of coding and reconstruction. Section V, presents results of experiments on a band-limited signals, and Section VI concludes with final remarks.

II. Signal Coding: An Artificial Spiking Neuron

Characterizing the general class of mappings from continuous time signals to spike trains is difficult owing to the

fact that there is no natural or canonical topology that one can impose on the space of spike trains. Simple subclasses, such as the set of all continuous mappings, therefore, can not be defined with an eye toward universal acceptance. A way out of this dilemma would be through the introduction of an intermediate continuous time signal that is transformed into a spike train via a simple stereotyped mapping—one where spikes mark threshold crossings. The complexity of the overall transformation can then be attributed to the mapping from the continuous time input signal to the continuous time intermediate signal. It is this approach that is followed here.

It is well known [2] that any time invariant, continuous, nonlinear operator with fading memory can be approximated by a finite Volterra series operator (assuming the uniform norm on the domain and range of functions). The general class of deterministic transformations from continuous time signals to spike trains can therefore be modeled as a cascade of a finite Volterra series operator which transforms the continuous time input signal into an intermediate continuous time signal, and a neuronal thresholding operation which marks the threshold crossings of the intermediate signal to generate a spike train. Here the simplest subclass of these transformations is considered: the case where the Volterra series operator has a single causal, bounded-time, linear term. This class is chosen with the tractability of the reconstruction problem in mind (see Section III). It is to be noted that the overall transformation from the input signal to the spike train remains nonlinear due to the thresholding operation.

To summarize, our interest lies in that subclass of transformations from continuous time signals to spike trains, that can be modeled as a cascade of a causal, linear, bounded-time filter, followed by a thresholding operation. This formal concept finds a natural home in a biologically plausible implementation of a neuron.

An artificial spiking neuron is a device which continuously measures a stimulus and marks the moments when the accumulated measure exceeds a certain threshold. The neuron does this via a convolution filter that is causal. Formally, let $k_n(t)$ be the causal convolution kernel associated with neuron n, and $t > 0$ denote the past. For a time window of size ℓ_n:

$$k_n(t) = 0 \quad \text{if} \quad t > \ell_n \text{ or } t < 0 \tag{1}$$

The measurement of the input signal $s(t)$ by neuron n at time t is $p(t)$ and is defined as:

$$p(t) = \int_0^{\ell_n} k_n(\tau) s(t+\tau) d\tau \tag{2}$$

Marking the moments where the measurement $p(t)$ crosses a threshold T is called spiking. A spike i is deemed to have occurred at time t_i if $p(t_i) = T$ and is non-decreasing at t_i .

With the addition of after-hyperpolarizing effects of previous spikes, the combined function

$$m(t) = p(t) + \sum_{i=1}^{M} \eta_n(t - t_i) \tag{3}$$

can be considered as a representation of the accumulated membrane potential of the cell. Here $\eta_n(t - t_i)$ is the after-hyperpolarizing effect of spike t_i.

Equivalently, a changing threshold can be used to simulate the after-hyperpolarization effect. Throughout the measurements and spiking, T is now not a constant but a variable $T(t)$. After a spike, T instantaneously rises and then decays exponentially. If neuron n produces M spikes until time t, then, threshold $T(t)$ for neuron n is expressed as:

$$T(t) = T_{init} + \sum_{i=1}^{M} T_J e^{r(t-t_i)} \quad T_{init} > 0, T_J > 1, r > 0 \tag{4}$$

T_{init} is the baseline threshold and T_J is the instantaneous increase in the threshold after a spike. r is the time constant which defines the rate at which the threshold decays back to baseline.

The aforementioned transformation from a continuous time signal $s(t)$, for $t > 0$, to a spike train $\langle t_1, t_2, t_3, \ldots \rangle$ can now be equivalently viewed as a corresponding neuron spiking at the respective times (that is, $p(t_i) = T(t_i)$ for all i) when presented with the signal $s(t)$ as input.

III. Signal Reconstruction

Given an ensemble of spike trains and characterizations of the corresponding set of neurons, one can ask whether the input signal can be recovered. This reconstruction problem is ideally posed as the inverse of the coding model. Each neuron model specifies constraints on the reconstructed signal that have to hold at the corresponding spike times. Anticipating that the formulation is under-constrained, among the many signals for which the constraints are valid, the one with the minimum energy can be chosen as the reconstruction of the original signal.

That convolutions can alternatively be viewed as inner products fully reveals the nature of this reconstruction. For each spike i, shifts $K_n^i(t) = k_n(t - t_i)$ can be defined where k_n is the convolution kernel associated with the neuron n and t_i is the time of the generation of spike i.

It follows that

$$p(t_i) = \int_{t_i}^{t_i+\ell_n} K_n^i(\tau) s(\tau) d\tau = \int_0^{\infty} K_n^i(\tau) s(\tau) d\tau \tag{5}$$

the last equality arising from the fact that $K_n^i(\tau)$ is 0 outside the range $[t_i, t_i + \ell_n]$.

A minimization problem can then be constructed as follows:

$$Min \ \|s\|_2^2 \ s.t. \tag{6}$$
$$T_n^i - \int_0^{\infty} K_n^i(\tau) s(\tau) d\tau = 0 \text{ for all } i \text{ and } n$$

where T_n^i denotes the threshold of neuron n at time t_i computed according to Equation 4. Although, for any moment in time, the absence of a spike can also be considered

as a constraint for the minimization problem, adding such constraints beforehand is impractical due to the reason that there are infinitely many such constraints. An alternative is to introduce such constraints in an incremental fashion based on the additional erroneous spikes that a candidate solution generates when coded by the same set of neuron kernels.

It follows from the *Representer Theorem* [6], that the minimum energy signal can be expressed as the weighted sum of the shifted kernels $K_n^i(t)$. The optimization problem can therefore be expressed in its dual form as a quadratic minimization problem with linear constraints:

$$Min \left\| \sum_{j,m} a_m^j K_m^j \right\|_2^2 \; s.t.; \qquad (7)$$
$$T_n^i - \langle K_n^i, \sum_{j,m} a_m^j K_m^j \rangle = 0 \;\text{ for all }\; i \;\text{ and }\; n$$

Relaxing the constraints with slack variables ξ_n^i one gets:

$$Min \sum_{i,n} \sum_{j,m} a_n^i a_m^j \left\langle K_n^i, K_m^j \right\rangle + c \sum_{i,n} \xi_n^i \; s.t.; \qquad (8)$$
$$\langle K_n^i, \sum_{j,m} a_m^j K_m^j \rangle - \xi_n^i - T_n^i \leq 0$$
$$T_n^i - \xi_n^i - \langle K_n^i, \sum_{j,m} a_m^j K_m^j \rangle \leq 0$$
$$\xi_n^i \geq 0,\; T_n^i \geq 0$$

Stated informally, given the neuron models, reconstructing a signal reduces to finding the best set of *weights* for the shifted kernels so that the linear combination of the shifted kernels creates the closest approximation of a signal that satisfies the threshold constraints.

Several observations are in order:

1) The convex optimization formulation of the reconstruction described above is valid for a coding model that contains only first order (linear) Volterra kernels. The reconstruction problem for higher order kernels does not naturally lend itself to a convex optimization formulation, although approximate reconstruction is possible.
2) The solution immediately reveals the trade-off between the length of a kernel l_n, and the sparsity of the spike code. Consider the schematic description of the solution to the reconstruction problem presented in Figure 1. If there is any inter-spike interval of time of duration larger than l_n, it follows immediately that the reconstruction drops to 0 for some intermediate period. Stated differently, the shorter the length of the kernel, the shorter the inter-spike interval needs to be in order for the reconstruction to be good.
3) On the other hand, a signal can be reconstructed only after l_n time has passed. This follows from the observation that the kernels of all spikes generated in the future within l_n time overlap with the present. l_n is therefore bounded from above by how quickly an organism needs to be able to reconstruct a signal.

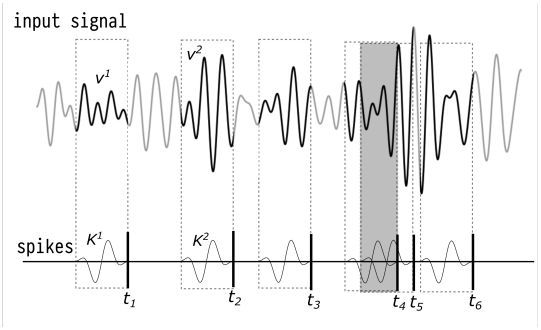

Fig. 1. Schematic diagram describing the reconstruction problem.

4) Finally, the quality of the reconstruction depends acutely on the shape of the kernel. This follows immediately from the fact that the reconstruction is a linear combination of shifted kernels. For example, a kernel situated in a lower frequency band than the input signal would fair very poorly at reconstructing the signal.

IV. Optimal Coding and Reconstruction

One is naturally led to ask whether, for a given class of continuous time signals to be coded, the kernels corresponding to the neuron models can be optimized to improve the quality of the reconstruction. This problem bears similarity to the sparse dictionary learning problem [7]. The kernels in our case correspond to the dictionary elements and the times of spikes relate to the sampling of the unknown signal. There is of course the additional complexity that kernels by design *overlap* in the reconstruction.

Under the (highly) simplifying assumption that the reconstruction does not involve kernel overlaps, even if the spikes themselves are not separated by the length of the kernel, an optimal set of kernels can be formally derived, as shown next. It is unfortunately the overlapping of the kernels that makes this an extremely challenging problem. This observation has also been corroborated in our numerical experiments.

For the sake of simplicity, a scenario where a single neuron kernel is derived, is considered. With repeated independent applications, the argument can be extended to the multiple neuron kernels case.

Let us assume that for a single neuron, $M = \langle t_0, \ldots, t_i, t_{i+1}, \ldots \rangle$ is a sequence of real numbers which represent hypothetical spike times.

Let $\tilde{s}$ be the optimal solution to the reconstruction of the input signal s. Appealing to the dual formulation in Eq 8, it is represented as:

$$\tilde{s} = \sum_i a^i K^i \qquad (9)$$

Note that, as before, for the spike i, the yet to be determined kernel k is shifted to time t_i and this shifted version is represented by K^i.

Rather than shifting the kernel to the locations of the spikes, one can alternatively extract segments of the signal aligned to

the spike times t_i. To elaborate, for the shift i, the section of input signal s which overlaps with K^i is represented by v^i.

$$v^i(t) = s(t + t_i) \quad for \quad 0 \leq t \leq \ell \tag{10}$$

Given any kernel k the reconstruction error is then given by:

$$\min_{a^i} \sum_i \left\| (v^i - a^i k) \right\|_2^2 \tag{11}$$

where the a^i's are free scalar variables that the above sum is to be minimized over.

Minimizing the above, the a^i's can be replaced with $\langle k, v^i \rangle$. Finally, assuming the bounding constraint $\langle k, k \rangle = 1$ and with further simplifications, one arrives at:

$$Max \ \sum_i (\langle v^i, k \rangle \langle v^i, k \rangle) \ s.t.; \quad \langle k, k \rangle = 1 \tag{12}$$

This, of course, is the well known principal component analysis problem and the optimum kernel is the first *Eigen vector* of the covariance matrix C built from the signal segments. Since the times of the spikes are unknown, all times may be considered to be equally probable in the construction of the covariance matrix. In conclusion, if it is assumed that the kernels do not overlap to represent the signal, the optimum kernel is extracted by selecting the first principal component of the covariance matrix generated from segments v^i sampled uniformly from the input signal s.

In spite of the questionable nature of the assumption that leads to the derivation of the above kernel, it was observed that the performance of the kernel is not entirely poor. Kernel derived with this approach is not a random waveform. For the sections of the input where spikes are sparse, the derived kernel will be scaled to give the best average representation. This requires it to be in the same frequency band as the input signal. And, for other sections, it is not going to be entirely irrelevant. For example, a kernel with frequency components outside of the frequency band of the input signal is highly unlikely to create a good representation. (Those frequency components need to cancel each other after shifting and scaling of the kernel at different positions.)

Returning to the general case as described in Eq 8 wherein overlap of kernels are allowed, the overall optimization problem reduces to one where the kernels K_n^i are also set to be variable (constrained of course by the fact that $K_n^i(t) = k_n(t - t_i)$ for a common kernel $k_n(\cdot)$). This unfortunately turns out to be non-convex. However, a procedure such as *block coordinate descent* [9] can be applied, the reason being that the problem is convex both when either the kernels K_n^i's are held fixed or when the coefficients a_n^i are held fixed. However, there is the added complication that changing the kernels K_n^i's leads to spike times that change which then changes the constraint set.

Considering the block coordinate descent step where the coefficients are held fixed and the kernels are optimized, further complications arise. Noting that the kernels are functions of time, optimizing the kernels at all time points is infeasible. However, one can choose a finite set of time points over which to optimize the kernels. This leads to the following block coordinate descent step.

Let $e(t) = s(t) - \tilde{s}(t)$ be the error between the input signal s and the reconstruction $\tilde{s}$ for any time t.

$$e(t) = s(t) - \sum_{i,n} a_n^i K_n^i(t) \tag{13}$$

For a set $M = \{t_0, \cdots, t_j, \cdots, t_N\}$ of time points,

$$\begin{aligned} Min \ \sum_j \xi_j \quad & s.t.; \\ & s(t_j) - \sum_{i,n} a_n^i K_n^i(t_j) - \xi_j < 0 \\ & \xi_j + s(t_j) - \sum_{i,n} a_n^i K_n^i(t_j) < 0 \\ & \xi_j > 0 \end{aligned} \tag{14}$$

For this step of the optimization problem, the weights a_n^i are fixed (from the previous step) and the values of $K_n^i(t_j)$ are optimized. With a sufficiently large set of time points M, kernels which reduce the total error can be obtained.

The overall procedure that has shown promise starts with a random distribution of spikes which don't cause kernels to overlap. For the non-overlapping case, kernels can easily be determined by solving constraint optimization 12. For this initial set of spikes and kernels, a reconstruction is obtained by solving optimization 8. This reconstruction becomes a reference and provides weights a_n^i for the next iteration of kernel optimization for which the equation 14 is solved. A new set of spikes are then generated with the updated kernels coding the input signal. This cycle is then repeated several times until a sufficient reconstruction performance achieved.

V. Experiments and Results

In this section, experiments on voice data are presented and their results are discussed.

Human voice was sampled at 8kHz and separated into a spectrum of frequency bands which are 100Hz wide. All tests were performed on the lowest frequency band(100Hz - 200Hz). Ideally, given enough time, same test can be performed for higher frequency bands as well.

Tests utilize a greedy algorithm which tries to find a good set of kernels by searching through random kernels. A random kernel can be created with the help of the special case described in section IV. As described in that section, placing random spikes which won't cause any overlaps, creates a special case for which it is possible to find the optimum kernel which minimizes the reconstruction error. The greedy approach creates many random kernels and forms a group from them. The group is modified as the search iterates. A random kernel is introduced to the group and for all the kernels in the group, the state of absence of each kernel is tested according

to its reconstruction performance. Briefly, a kernel is replaced with the new one if the reconstruction performance of the group improves. This algorithm can be altered by introducing a kernel optimization routine by implementing the problem described by the equation 14. Instead of creating a totally random kernel, a kernel which is already in the group can be altered by the optimization routine and the performance of the group can be tested. It is already stated that although it is possible to partially alter the kernels, it does not always improved the reconstruction. This is due to the fact that, altered kernel invalidates the spike positions. Kernel optimization is a relatively time consuming job, because of that reason, it is not included in results presented here.

The input signal was divided into sections. Although there is no hard constraint on the length of a section, longer sections will create relatively harder optimization problems with higher number of parameters. For the tests presented in this paper, the section length was selected to be approximately 0.5 seconds or 4000 samples. Deciding to solve for very short duration of the input can introduce problems when it comes to generalizing the selected kernels to the other sections of the input. There is a trade-off between the representation accuracy for a section and success in generalizing that representation to other sections. A similar argument can also be made for the length of the kernels. Relatively short kernels create more accurate representations but they are required to be placed high in numbers. For the test presented in this paper kernels were chosen to be equal in length(160 samples).

Any reconstruction which is not exact, will create a non-zero error signal. In order to get a better reconstruction, this error signal can be threated as a regular signal and a reconstruction of it can be calculated. A staged structure can be created by creating an additional step of calculating the error and trying to represent this error with spikes and kernels. With this staged structure, the idea of representing a signal with spikes and kernels will still be in focus. With sufficient abstraction, this approach of creating a staged or layered structure can be explained as follows. A group of neurons take the input signal and create spikes. These spikes and the kernels of the neurons are then used in order to calculate a reconstruction of the input signal. Then, the reconstruction error is calculated. A second group of neurons take this reconstruction error and generated spikes from it. It is not possible to create a biological analogy to this staged structure. That is because there isn't any explicit evidence of a reconstruction or an error calculation is observed in nature. Hypothetically, this can be analogues to a case where a neuron creates spikes from input signal and a second neuron takes these spikes as auxiliary input along with the input signal. In order to calculate a difference signal, the synapse strength connection two neurons should be adaptive and continuously changing.

More stages can be added if further refinement of the reconstruction is desired. Figure 2 gives a graphical demonstration for the 2 stage version of the idea of creating multiple stages.

Adding more stages increases the quality of the reconstruction but at the same time it decreases the efficiency of the

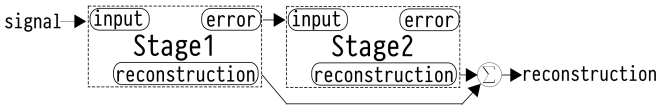

Fig. 2. Expanding and improving the overall reconstruction with multiple stages of residual representations

A stage accepts a signal and calculates a reconstruction. The reconstruction error can also be represented by spikes and kernels. So, with multiple stages, it is possible to improve overall reconstruction accuracy.

representation.

Although, currently, there does not exist any known method of deriving the optimum set of kernels, tests show that some sets of kernels perform better than others. And, with the help of a greedy search a more efficient representation can be achieved.

Regardless of the decided kernel length or the section length, different set of kernels show different reconstruction performances. Figure 3 shows the results of greedy algorithm and random selection of kernels for each stage. A section from the input signal is selected and the greedy algorithm tries to find "good" kernels for each stage by checking their reconstruction performance for the section selected. Then, these kernels are tested on other sections of the input signal. The performance of the selected kernels is not as good as the case when they are applied to the initially selected section.

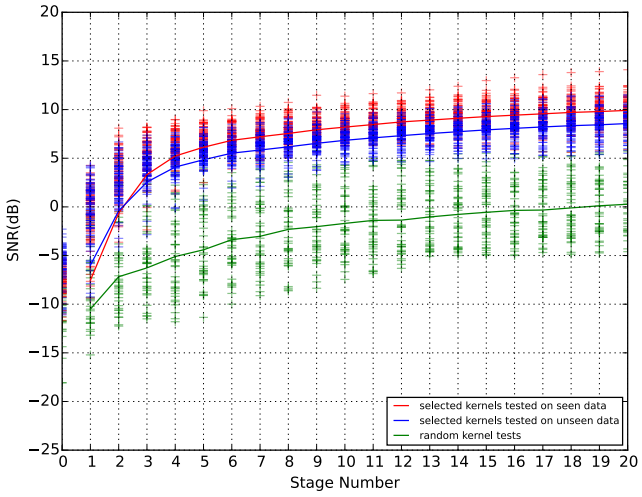

Fig. 3. Reconstruction Errors at Different Stages

Each '+' sign represents a reconstruction test. Plot shows that as the number of stages increases, signal to noise ratio improves. Blue signs are for the test on unseen sections of the input. Red signs represents tests related with the sections which are chosen as reference for kernel selection. Green signs represent tests with randomly chosen kernels.

Since, in order to find better kernels, the greedy algorithm tests random kernels over and over again, it is expected that

more iterations of the algorithm will end up with better kernels. Figure 4 gives an idea about how likely it is to get better kernels with deeper searches. According to the tests, deep searches which have more than 50 kernel replacements in the group did not show significant improvement in reconstruction performance.

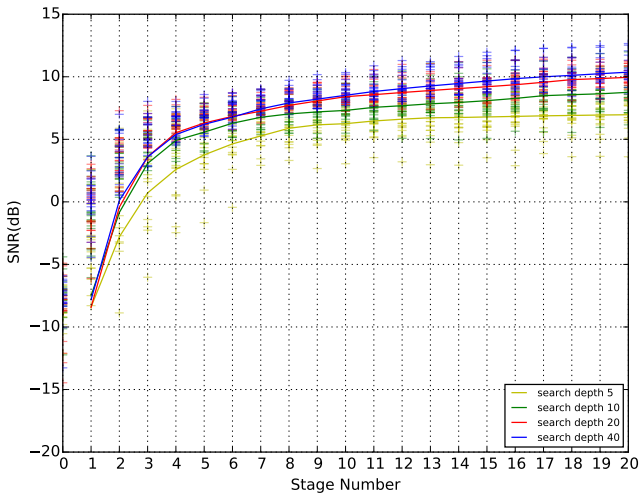

Fig. 4. The Effect of Search Depth on SNR
Each '+' sign represents a reconstruction test. Four different search depths are compared. A search depth of 5 means that the greedy search algorithm searched and replaced 5 kernels. As expected, the more it runs the better kernels it finds. The relative improvement decreases as the depth of search increases. There isn't much difference between a depth 20 search and a depth 40 search.

Tests were repeated 80 times for the section which is chosen as a base for kernel selection. For these tests the average Signal to Noise Ratio was slightly above 10dB. SNR for unseen sections of the input was above 8dB. Perception of the quality may not be entirely linked to the SNR of the coded signal but it still gives an idea about the quality of the coded signal. According a relevant study [1], a speech signal with SNR greater than 10dB can be tolerated or understandable.

Although, each run of the greedy search algorithm creates a different sets of kernels, seeing an actual reconstruction of an input section together with some of the selected kernels can still give some sensible information about the approach introduced in this paper. Figure 5 displays the results of a sample run. For this sample run, 3 neurons were selected for each stage. At each stage, 3 kernels produced around 40 spikes in total. Although for a single section, the amount of data needed for such representation seems high, same kernels can be used for other sections and efficiency for the overall representation decreases. Key frames or key sections can be defined throughout the input and generalizing kernels based on a single section can be relaxed to some extend. Kernels (Figure 6) shaped such that it looks impossible to parametrize them and create a filter-bank in order to generate similar kernels. Different from gamma-tone filters, chosen kernels have more complex forms. With choosing kernels such, a more compact but approximate representation of a signal is aimed.

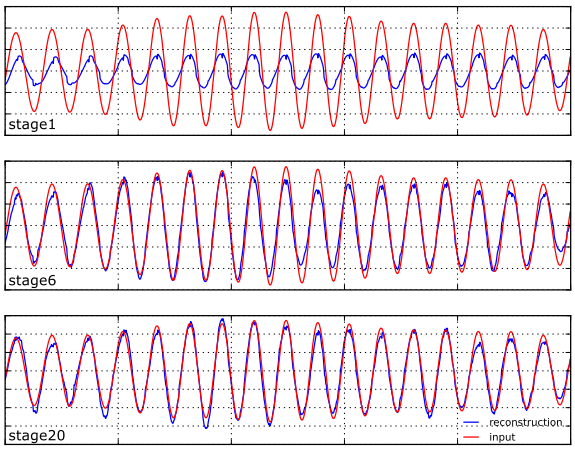

Fig. 5. Reconstruction at Different Stages
Three stages (1, 6 and 20) selected in order to show how the reconstruction improves. There isn't much difference between stage 6 and the stage 20. Sections where the amplitude is relatively high can reach an adequate representation at early stages due to the abondance of the spikes. For a different section of the input this may not be the case.

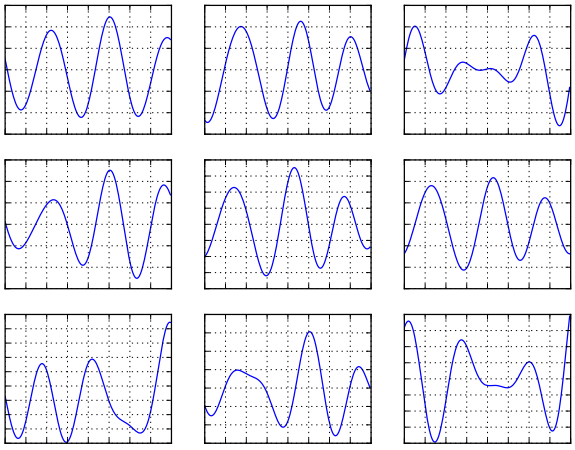

Fig. 6. Kernels at Different Stages
Three stages (1, 6 and 20) selected in order to show how the shapes of kernels used. They don't have any predictable shape or form.

VI. Conclusion

In this paper, a model aware approach of coding a continuous time signal with an ensemble spike trains has been proposed. Spikes generated through a model is assumed to be a representation of a signal. The definition of the model together with the generated spikes define a coding of the signal. With the approach presented in this paper, a representation of the signal can be traced back to the signal itself as a reconstruction by utilizing both the generated spikes and the definition of the model which generates the spikes. A definition of the reconstruction error based on the coding of the reconstruction makes it possible to assess the relevance of model to the objective of creating better reconstructions. Through such an assessment procedure, suitable models for the objective in question can be searched for. The approach has been tested on voice data and promising results have been observed. Successively expanding the representation of the input signal by including the spike based coding of the reconstruction error, made it possible to get a monotonic improvement on the representation accuracy. This paper shows that with a deterministic model, it is possible to get a reasonably good reconstruction of the input signal. It is pointed out that currently there does not exist any proposed method of finding kernels which give the best reconstruction performance with maximum efficiency. Future research will be on predicting the shapes of kernels by modifying the spike positions. Experiments, which are at their early stages, show that it may be possible to find kernels which consistently produce the spikes at certain positions. Instead of randomly placing spikes and creating random kernels, spikes can be placed at certain positions and kernels can be aimed to represent sections of the input which are expressed relatively poor. With any modification to kernel finding process, the proposed idea of representing the signal with spikes and kernels will still be valid and provide a basis for future research.

Acknowledgment

The authors would like to thank the Air Force Office of Scientific Research (Grant FA9550-16-1-0135) for their generous support of this research.

References

[1] Bishnu S Atal. Speech coding: recognizing what we do not hear in speech. *Annals of the New York Academy of Sciences*, 405(1):18–32, 1983.

[2] Stephen Boyd and Leon O Chua. Fading memory and the problem of approximating nonlinear operators with volterra series. *Circuits and Systems, IEEE Transactions on*, 32(11):1150–1161, 1985.

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

[4] Ilya Nemenman, Geoffrey D Lewen, William Bialek, and Rob R de Ruyter van Steveninck. Neural coding of natural stimuli: information at sub-millisecond resolution. *PLoS Comput Biol*, 4(3):e1000025, 2008.

[5] Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

[6] Bernhard Schölkopf, Ralf Herbrich, and Alex J Smola. A generalized representer theorem. In *Computational learning theory*, pages 416–426. Springer, 2001.

[7] Andreas M Tillmann. On the computational intractability of exact and approximate dictionary learning. *Signal Processing Letters, IEEE*, 22(1):45–49, 2015.

[8] Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005.

[9] Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.