TempUnit: A bio-inspired neural network model for signal processing

Olivier FL. Manette, Marc A. Maier

Abstract – **We have developed and tested a novel artificial neural network for the processing of temporal signals. The working of the units (TempUnit) is based on the mechanism of temporal summation as observed in biological neurons. Its particularity is to adapt its basis function by supervised learning. The model was tested on cortical and associated muscular (EMG) recordings from the behaving primate. The TempUnit showed a 2.3 times better performance in mapping spiking to EMG activity than a time delay multi-layer perceptron. Data compression capacity of the TempUnit was tested on audio data and compared to the MP3 compression standard. For a similar reproduction quality, we found a compression rate 5 times higher than in MP3.**

I. INTRODUCTION

Most artificial neural networks use predefined, fixed and analytical basis functions for learning and generalization. However, the capacity for learning and generalization depends on the choice of the basis function. Often, the choice of the basis function is inspired by transfer functions observed in biological systems.

There are at least two distinct classes of neural transfer function and coding schemes: value coding and intensity coding [1]. Value coding is defined by a neural response that codes for a specific value of a parameter, such as the orientation of a straight line in the receptive field of a visual neuron, or the direction of a limb movement in space. Often, the basis function associated with value coding is a Gaussian [2-11]. The second class concerns intensity coding, where a given range of the parameter is coded, usually by linear or sigmoid-like basis functions [12, 13]. The computational advantages of those two coding schemes have been investigated through the use of artificial neural networks [14] and found to have particular properties in terms of learning and generalization [14-16]. Furthermore, and more specifically for temporal processing, more complex usually analytical and fixed basis functions are used such as in Fourier transform (sine and cosine) and wavelet transform.

Here we investigate an artificial neural network that is not based on a fixed basis function, as in the models mentioned above, but that learns an optimal basis function to map the input to the output. Optimizing the basis function on the given data should allow faster learning and better generalization than the use of fixed and predefined basis functions.

We will use artificial and biological data to test our neural network, which is based on a formal model of temporal summation (TempUnit) and on information processing of time-varying signals. In the biological domain, we will explore the relation between spike trains (input) and EMG (output). Within this framework, the following two questions will be addressed: i) how is the output EMG determined by the input spike train (direct computation)? ii) Given an EMG profile or any other target signal, how can the input signal be determined (inverse computation) that leads to the output profile?

A single TempUnit (equation 3) in its most simple form, without temporal scaling (dilatation), is comparable to a Finite Impulse Response (FIR) filter. MLP with FIR filter synapses (FIRNN) have already been developed for time series predictions [17-19]. These FIRNNs are functionally equivalent to time delay multi-layer perceptrons (TDMLP), considering that each FIR filter coefficient corresponds to a static synaptic weight of a TDMLP unit [20, 21]. Hence, the learning of the FIR filter coefficients can be achieved by algorithms similar to temporal backpropagation [22, 23].

We will demonstrate supervised learning in TempUnit networks and compare their performance to those of other networks such as TDMLPs. Furthermore, we will compare the performance of TempUnit networks in terms of data decomposition and data compression with that of wavelet transforms (with MP3 algorithm).

II. TEMPUNIT: A MODEL BASED ON TEMPORAL SUMMATION

The TempUnit model is based on the mechanism of temporal summation of postsynaptic potentials observed in neurons, i.e. every time a spike arrives at a synapse, it triggers a local (postsynaptic) potential, and when many spikes arrive during a short period of time, these potentials add up. In particular, the resulting potential is much larger for near coincident arrival of spikes [24, 25], which led to the formulation of theories and mechanisms of temporal coding in the brain [26-30]. Empirical data indicate that neurons, when driven by fluctuating stimuli, respond in a deterministic fashion [31] as does the TempUnit.

Manuscript received February 14, 2006. This work was supported in part by the European 5th framework programme: IST-2001-33073 and by a French grant: ACI Neurosciences intégratives et computationnelles 02-2-0667.

Olivier FL. Manette is with Inserm, U742, Paris, F-75005, France ; Université Pierre et Marie Curie (Paris-6), Paris, F-75005, France (e-mail: olivier.manette@free.fr).

Marc A. Maier is with Inserm, U742, Paris, F-75005, France ; Université Pierre et Marie Curie (Paris-6), Paris, F-75005, France (tel: 0033 1 44 27 37 47; e-mail: Marc.Maier@snv.jussieu.fr).

The TempUnit thus provide a binary-to-analog conversion (and vice-versa). The temporal and deterministic evolution of the membrane potential **r** of the TempUnit is the result of temporal summation and depends on 2 parameters. First, spiking input activity, given in vector **x**, and second on the shape of the local potential induced by a single spike, i.e. on a basis function defined by vector **v**. In the context of discrete time, vector **v** has a duration of *p* bins. We can hence define the binary input activity as a vector $\mathbf{x}_t$, for *p* bins (eq. 1), where 1 indicates the presence and 0 the absence of a spike. Vector **x** corresponds to a moving window of size *p* that slides along the entire spike train x_t of size *T*. However, in order to include the possibility of temporal scaling of the output, we introduce a coefficient for dilatation (*Cd*). The TempUnit is thus able to scale in time a temporal pattern, as has in some cases been observed for the EMG [32]. *Cd* is expressed as the ratio of the sampling rate of the input to the sampling rate of its basis function vector **v**. In equation 2 we express the integer part as $\mathrm{E}(x) = \left[x \right]$ and the decimal fraction as $x - \mathrm{E}(x) = \lfloor x \rfloor$. The resulting potential $\mathbf{r_t}$ at time *t* is then given by equation (2). Thus, the spikes in the period *p* prior to *t* are necessary and sufficient to determine the potential **r** at *t* (Fig. 1).

$$\mathbf{x}_t = \left[x_{t-p-1} \ldots x_{t-1} \right] \quad \text{(columns } 1 \ldots p\text{)} \qquad (1)$$

$$r_t = k + \sum_{i=1}^{\left[Cd \cdot p \right]} x_{t-i-1} \left(v_{\left[\frac{i}{Cd} \right]} + \left\lfloor \frac{i}{Cd} \right\rfloor \left(v_{\left[\frac{i}{Cd} \right]+1} - v_{\left[\frac{i}{Cd} \right]} \right) \right) \qquad (2)$$

In the particular case of *Cd*=1 we get :

$$r_t = k + \mathbf{x}_t \cdot \mathbf{v} \qquad (3)$$

In the remaining part of this article, we will use *Cd*=1.

The output **f** of the TempUnit is then calculated by combining **r** with the synaptic weight *w* and with the bias *k*. Several TempUnits can be combined to form a feedforward neural network (Fig. 2). Every neuron *j* of the *N* neurons in this network has its own basis function $\mathbf{v}^j$, synaptic weight w^j and bias k^j.

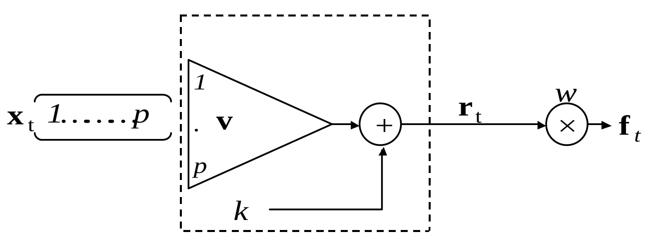

Figure 1: A schematic representation of the TempUnit. $\mathbf{x}_t$: binary input, **v**: basis function, $\mathbf{r}_t$ analog output, *w*: weight, *k*: bias, $\mathbf{f}_t$: weighted analog output.

In order to have a complete input-output relationship for the entire spike train, we form the matrix $\mathbf{X}^j$, where every row corresponds to a vector $\mathbf{x}_t$. The complete output **f** of the network is described by equation (4):

$$\mathbf{f} = \sum_{j=1}^{N} \left(\mathbf{X}^j \cdot \mathbf{v}^j + k^j \right) w^j \qquad (4)$$

In terms of signal processing, the TempUnit can be compared to other models using basis functions to decompose a signal, such as the FFT or wavelet transform. However, on a biological level, the TempUnit forms a computational substrate that allows the processing and analysis of binary, temporal information. Of course, other models also allow for such an analysis (e.g. [16]), however, these models incorporate fixed and predefined basis functions, whereas the TempUnit incorporates a basis function acquired and optimized through supervised learning, as described below.

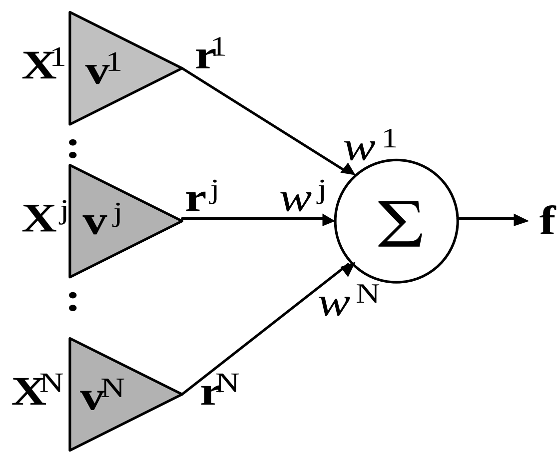

Figure 2: Feed-forward network of N TempUnits. Every gray triangle corresponds to the stippled box in Fig. 1.

III. Supervised Learning

In the specific case of a dilatation coefficient *Cd*=1 the TempUnit is equivalent to a FIR filter operating on a binary input and thus the proposed learning algorithm may also be applied to certain classes of FIR filters.

Usually, supervised learning in neural networks optimizes the weights and the units classically use fixed transfer functions. In contrast, supervised learning in the TempUnit serves to determine an optimal basis function. This requires the input (spike train) as well as the target output (EMG). The input is directly used to determine the position and combination of the basis function (eq. 2 and 3). The output, in conjunction with the time-linked input, is used to determine the optimal form of the basis function through supervised learning. In addition, the duration of the basis function needs to be pre-specified. Thus, the input consists of the binary matrix **X** of dimension *p* times *T-p*, i.e. the row-vector x time-shifted by one bin per time step over the entire spike train. This provides an overdetermined system of *T-p* linear equations with *p* unknowns and *T* inputs (*T* > *p*). The compromise solution of the linear system is then determined by minimizing the differences.

We will illustrate learning and performance of a TempUnit, first, by learning a sinusoidal signal and second, by the use of biological (cortical and EMG) data obtained in the behaving monkey.

A. *Sinus function*

The target output consists of 2 cycles of a sine wave sampled over 32 bins. For determining the basis function we also need the input function, which shall consist of a spike train of 8 consecutive spikes (p=8) centered over the peak (Fig. 3). The matrix **X** thus is of dimension 8 x 24 and the system of 24 linear equations with 8 unknowns can be solved.

With the basis function shown in Fig. 3, the TempUnit reproduced the 32 samples of the sine wave with a performance close to 1 (P=99,999%; error=15.76e-6). Performance was calculated as P=1-(error/$error_{max}$), whereby $error_{max}$ = average signal – target signal (see also [33]).

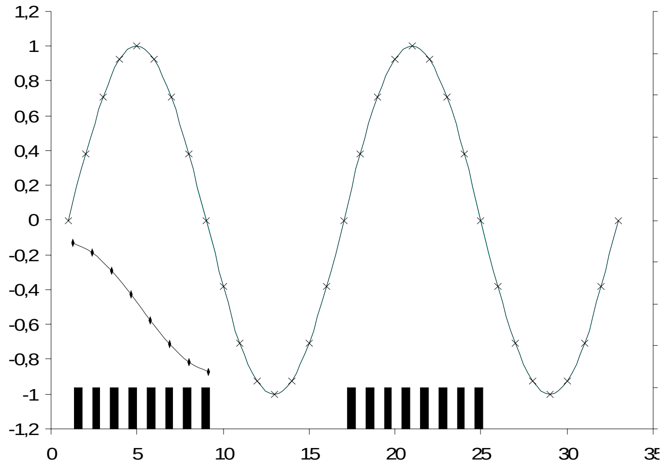

Figure 3: Sine wave example. Vertical bars: input spikes. Solid line: target output. Crosses: computed output. Solid line above the first eight spikes: learned basis function.

B. *Muscle and brain activity*

In the brain, there are several different populations of neurons that activate the motoneurons [34], among them the cortico-motoneuronal (CM) cells located in the primary motor cortex [35, 36]. The activity of the motoneurons then produces muscular (EMG) activity. Thus, CM cells are causally involved in the production of EMG activity and it is thus of particular interest to determine the transfer function between the activity of CM cells and the resulting muscle activity. For example, this kind of transfer function might be used in the control of prosthesis, for functional electrical stimulation or brain-computer interfaces. For our purpose, we used simultaneously recorded CM cell and target EMG data from the behaving monkey [37]. Contrary to the case of the sine wave illustrated above, here, both the input and the desired output are known. Thus, the straightforward application of the learning algorithm will yield the basis function. Fig. 4 shows the resulting basis function for one example cell (CM06). The U-shaped basis function is compatible with the two distinct periods in the CM cell spike train, where frequency and temporal coding might occur [33]. It also contains higher frequency components most likely linked to the motor unit potentials seen in the target EMG. Prediction of the time-varying EMG over a period of 400 s by the TempUnit using the learned basis function and the original spike train provided a performance of P=26.11%. Given that the prediction was based on a single CM cell, this is a quite good performance, considering that the EMG activity is determined by the combined activity of many CM cells and other neurons.

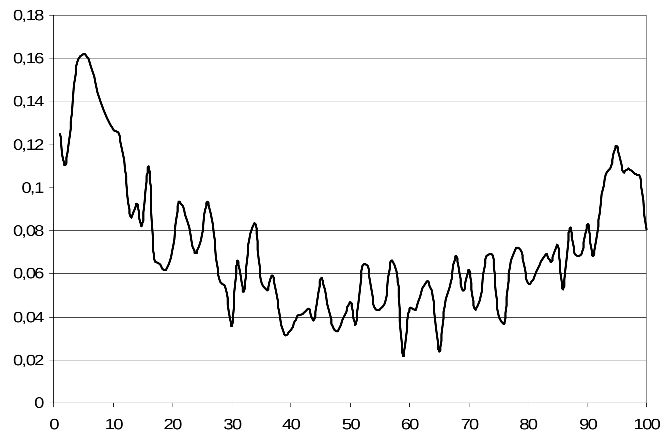

Figure 4. The learned basis function **v** between the brain (CM cell) activity, and a muscle (EMG) activity. p=100 bins (equivalent to 400 ms).

C. *Comparison with a time delay Multi-Layer Perceptron*

We have previously investigated the same transfer function between CM cell and muscle activity [23] with the help of supervised learning in a time delay multi-layer perceptron (TDMLP). We can thus compare the performances of this TDMLP (100 input units, 15 units with sigmoïdal transfer functions in its hidden layer, 1 output unit with a linear transfer function; with a total of 1515 synaptic weights to be determined) against that of a single TempUnit with the same input (i.e. p=100 with a total of only 100 parameters to be determined). To obtain the best TDMLP performance using a back-propagation learning rule, 100 learning sessions were performed after a random Nguyen-Widrow initialization of the weights.

Based on the activity of 24 CM cells and their corresponding target EMG, the average performance of the input-output mapping was 8.90% for TDMLP and 29.37% for the TempUnit. On those biological data, the TempUnit was far better than the TDMLP. A likely explanation for the better performance of the TempUnit resides in its biologically inspired mechanism of temporal summation.

In addition, and unlike the TDMLP that acts as a black box, the TempUnit provides an explicit basis function (Fig. 4). Furthermore, with a TDMLP it is very difficult to define an inverse function, whereas this is possible with the TempUnit, as explained below.

IV. Inverse functions

A. Inverse function

In some applications, and specifically in motor control, it is necessary to determine in advance which kind of input (e.g. motor command) is able to produce a specified output (e.g. movement trajectory). The TempUnit model allows us to determine the correct input for a desired output. Since the output of a TempUnit is a linear combination of the input **x** and of the basis function **v**, it is straightforward to resolve the input if the output and the basis function **v** are given.

To obtain the inverse function, we formulate a linear system of equations and inequalities such that its solution provides the inputs x_t. We need at least p consecutive output values v_i to resolve the system. With those values we can write p-1 equations (5) and 1 equation (6):

$$r_{t-j} = \sum_{i=1}^{p-j} x_{t-i-1} v_{i+j} + K_j \text{, for } j\text{=1 to } p\text{-1} \quad (5)$$

$$r_t = \sum_{i=1}^{p} x_{t-i-1} v_i \quad (6)$$

In equations (5) and (6) we need to determine u_i and K_j. We have hence only p equations for 2p-1 unknowns. However, by writing inequalities in order to get more equations we will be able to solve the system. So, if the input is a binary, then:

$$0 \le u_i \le 1 \text{, for } i\text{=1 to } p \quad (7)$$

and

$$0 \le K_j \le \sum_{i=1}^{j} v_i \text{, for } j\text{=1 to } p\text{-1} \quad (8)$$

This results in a solvable system of 2p-1 equations and inequalities for a total of 2p-1 unknowns.

B. Interpolated inverse function

In some applications of inverse functions, only the initial and the final state are known but not the intermediate states. Inverse calculation should provide a solution for determining these intermediate states, as for example in motor control for the trajectory formation from an initial point A to a final point B. The same problem holds at the level of the EMG or at the level of neuronal activity. We will explain an algorithm for interpolated inverse functions of the TempUnit in the domain of spiking neurons, i.e. how to get from a spike train A, representing the initial state, to a spike train B, representing the final state. However, if the initial and final state is given in the domain of the EMG, then we can use the regular inverse function to calculate the corresponding spike trains.

A and B need to be represented in a common coordinate system in order to calculate the distance between A and B (Fig. 5). Then, at each time step a state transition needs to be determined, such that the distance between the actual state and the final state decreases. For a spiking neuron, this means to determine whether it should spike or not at every time step, such that, at the end, the spike train reaches its final state B, starting out from state A.

Since a train of binary events can be described in the frequency as well as in the temporal domain, we will explain the evolution of these two aspects in a two-dimensional coordinate system. Indeed, in biological spike trains, frequency and temporal coding has been documented in many instances (e.g. [38]). In the following, every input vector $\mathbf{x}_t$ has a specific coordinate, corresponding to its frequency and temporal aspect. Given the initial state $\mathbf{X}_A$, expressed by its coordinates ϕ_A and τ_A:

$$\mathbf{x}_A = \{\phi_A ; \tau_A\} \quad (9)$$

we can then calculate ϕ_A, expressing the average frequency:

$$\phi_A = \sum_{i=1}^{p} x_{t-i-1} \quad (10)$$

and τ_A expressing the temporal coding:

$$\tau_A = \sum_{i=1}^{p} 2^i x_{t-i-1,i} \quad (11)$$

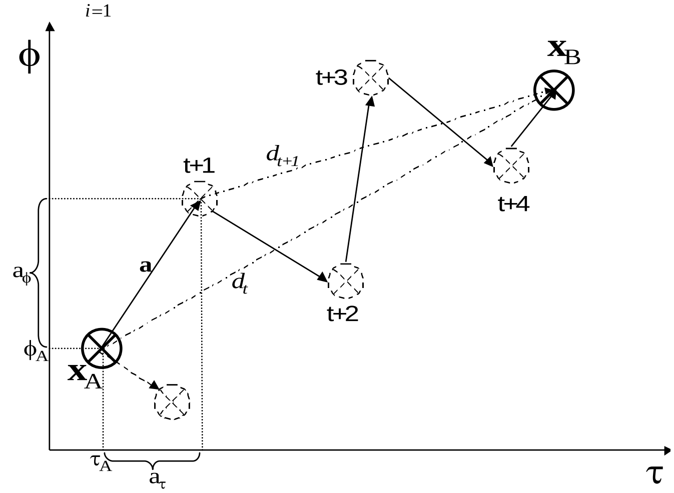

Figure 5: A schematic illustration of the algorithm for the interpolated inverse function. X_A: initial state, X_B: final state. The intermediate states are reached at time t=t+1, t+2, t+3 and t+4.

After having calculated the coordinates for both A and B, the distance between A and B can be determined. For each transition (time step), we determine whether the emission of a spike or the absence of a spike will minimize the distance to B. The transition vector **a** has the coordinates a_ϕ and a_τ that can be calculated for the frequency component:

$$\mathbf{a}_\phi = \phi_{t+1} - \phi_t = x_t - x_{t-p-1} \quad (12)$$

and for the temporal part:

$$\mathbf{a}_\tau = \tau_{t+1} - \tau_t = \sum_{i=1}^{p-1} 2^i \left(x_{t-i} - x_{t-i-1}\right) + 2^i \left(x_t - x_{t-1}\right) \quad (13)$$

The activity of the neuron at the new time step (x_t) is the only unknown in equations (12) and (13). The new activity x_t of the input neuron is calculated by minimizing the distance d to B (14):

$$d = \sqrt{\left(\phi_A - \phi_B + \mathbf{a}_\phi\right)^2 + \left(\tau_A - \tau_B + \mathbf{a}_\tau\right)^2} \qquad (14)$$

This procedure, applied iteratively until d=0, will lead to B. However, contrary to the regular inverse function, this heuristic does not guarantee a solution in all cases.

In the following, we will apply the regular inverse function and supervised learning to data compression.

V. DATA COMPRESSION

Data compression has become a major issue in signal processing, especially for efficient data transmission and storage but also in computational intelligence. In particular, it might be more efficient to store the information of how a signal might be recomposed, rather than storing the full signal itself. Somewhat similarly, an analog EMG profile might be stored by multiple binary spike trains. Usually, in signal decomposition, the position, size and combination of a given basis function is determined independent of the input. In our case, for optimizing the basis function, both, the input and the output are required. However, if there is no input-output mapping available, the input can be generated. We will illustrate both cases and first investigate whether data compression occurs between cortical and EMG signals for which the basis functions have already been determined. Second, for illustrating the compression of an audio signal, where no spike-to-analog mapping is available, we will generate a spike train that subsequently allows for an iterative optimization of the basis function.

A. EMG activity

The memory space for storing a CM cell spike train and its basis function, which are needed to recompose the EMG signal, can be compared to that needed for storing the original EMG signal. For this example, three simultaneously recorded CM cells (CM04, CM05 and CM06) having the same target muscle (Abductor Pollicis Brevis) were used to calculate three independent basis functions. These three basis functions (i.e. three TempUnits arranged as showed in Fig. 2) were used to decompose the target EMG by the use of the inverse algorithm. Subsequently, three artificial spike trains (Fig. 6B) were generated to determine the position and combination of the underlying basis functions to obtain an artificial EMG. Figure 6A shows the recorded EMG activity (gray trace) over a period of 20s and the corresponding TempUnit output (the artificial EMG, black trace). The three learned basis functions are shown in Fig. 6C.

Comparing the original, recorded EMG to the computed, artificial EMG resulted in a network performance of P=64%. The performance taking into account each single TempUnit was 34.1%, 25.7% and 26.1% for CM04, CM05 and CM06 respectively. The sum of the independent performance is larger than the network performance. This suggests that individual CM cells fire in a redundant manner. This redundancy is also expressed by populations of CM cells that show similar firing patterns [13]. In addition, the artificial spike trains showed a clear resemblance to biological activity: one train is more phasic, the other more tonic, patterns which have been found in-vivo [13].

For this period of 20s the inverse algorithm computed a total of 2878 spikes and obtained a performance of P=64%. The storage of those 2878 spikes and their basis functions corresponds to a compression rate of 3.1:1.

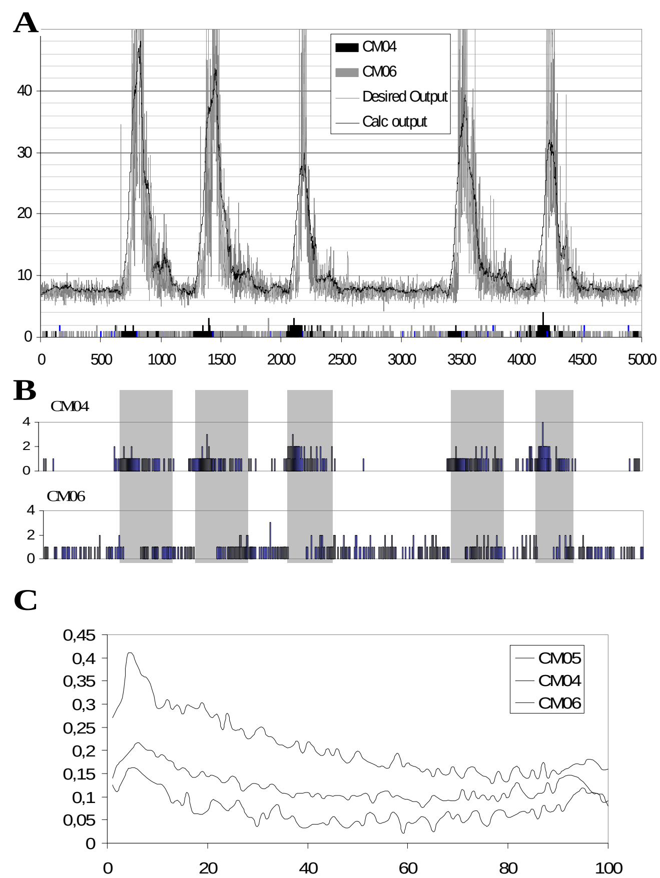

Figure 6: Forward and inverse calculation of biological data. A. Recorded EMG traced showing five tasks-related bursts (gray line). TempUnit output (solid line). Bottom: spiking activity of the three TempUnits. B. Artificial spike trains calculated by the inverse function for two of the three TempUnits. The third TempUnit had very few spikes and was omitted from the figure. Shaded areas indicate period of phasic activity in the EMG. Note that CM04 shows a phasic activity, whereas CM06 a rather tonic activity. C. Basis functions for each of the three TempUnits determined by supervised learning.

B. Audio signal

We tested the compression rate of the TempUnit on 108s of audio data sampled at 44.1 kHz from a Maceo Parker CD. Part of the audio signal is shown in Fig. 7A. We used a semi-supervised learning scheme to code the audio signal through an artificial spike train and its associated and learned basis function: first, spikes were placed according to the derivative of the audio signal, i.e. spikes of high frequency for periods of large positive or negative derivatives and no spikes for small derivatives. Then, iteratively, a basis function was determined by the supervised learning procedure described in chap. III, followed by the inverse function to determine a more accurate spike train until learning saturated.

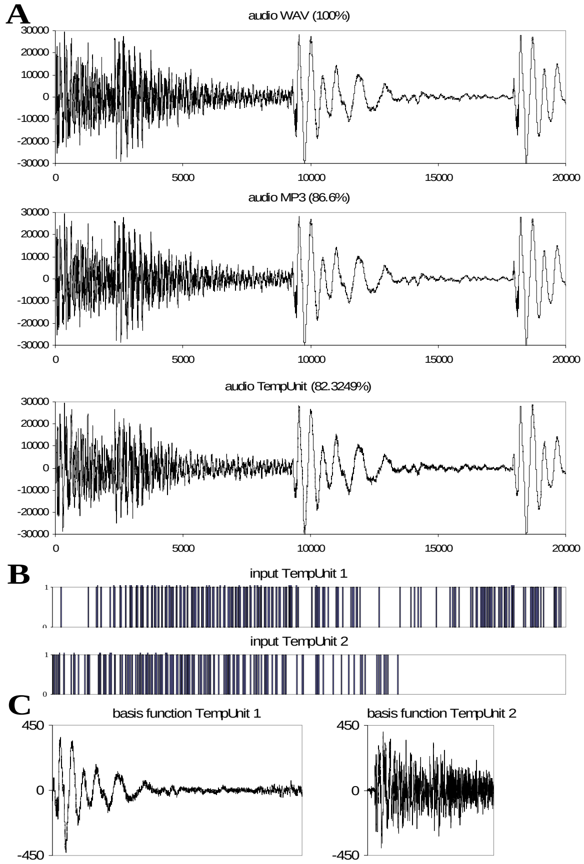

Figure 7: Data compression of an audio signal through a network of two TempUnits. A. From top to bottom: original signal (0.45s are shown), restored MP3 signal, restored TempUnit network signal. B. Artificial spike train for the two TempUnits over the same period of 0.45s. C. Basis functions for the two TempUnits.

Fig. 7B shows the reconstructed artificial spike trains and Fig. 7C shows the two basis functions. The first basis function was chosen to be twice as long as the second. The first captured a signal of lower frequency content than the second. This suggests that the length of the basis function is related to the frequency content of the signal. The performance rate (inverse of the error) of the TempUnit was similar to that for MP3 at 56Kbps (Table 1).

TABLE 1: COMPARISON OF PERFORMANCE AND COMPRESSION RATE BETWEEN MP3 AND TEMPUNIT ALGORTIHMS

	Performance	Compression rate
Original data	100%	1:1 (18.2 MB)
MP3	86%	1:12.6 (1.49 MB)
TempUnit	83%	1:70 (266.5 KB)

However, the compression rate of the TempUnit was 5 times higher. This indicates that, relative to the performance, the compression rate of the TempUnit is superior to that of MP3. However, since the compression rate of the TempUnit depends on the complexity of the signal, these comparative data, although based on an identical source, might only be approximate.

The file for the recomposition of 83% of the audio signal contains the two spike trains as well as the two corresponding basis functions. One basis function over 10000 bins with a 16 bit resolution (20kB), another basis function of 5000 bins (10kB), 80125 spikes for the first and 41343 for the second TempUnit (16 bit / spike = 237kB), i.e. a total of 267kB.

VI. CONCLUSION

The feed-forward TempUnit network shows a particular capacity for the processing of temporal signals. It is based on the mechanism of temporal summation as observed in biological neurons. For testing the model we used cortical activity associated to EMG activity obtained from the behaving primate. Compared to a time delay multi-layer perceptron, the TempUnit showed a 2.3 times better performance in mapping spiking to EMG activity. We then tested the compression rate between brain and EMG signal and found a rate of 1:3 with a biologically compatible basis function. This compression rate is relatively low and might be a consequence of the U-shaped basis function, which may reflect biological mechanisms or constraints that are not optimal for data compression. However, since populations of neurons with relatively redundant firing patterns activate a given muscle, the actual compression rate is likely to be higher. To test the compression capacity of the TempUnit without any biological constraints, we then used an audio signal and compared the compression rate to the MP3 standard. For a similar reproduction quality, we found a compression rate 5 times higher than for MP3.

The specificity and novelty of the TempUnit concerns i) its binary-to-analog transform of time-varying signals, ii) its use of temporal summation of incoming signals, iii) the capacity for temporal scaling (dilatation), iv) its capacity to acquire a basis function through supervised learning based on linear system equations which then allows for fast learning based on a single cycle; v) the possibility of an inverse calculation. Because the TempUnit allows for inverse calculation it can also be used to predict spike trains from analog (EMG or for example sub-threshold intracellular) signals. In short, the TempUnit combines certain aspects of wavelet transforms (temporal scaling) with those of FIR filters (non-analytical, optimized basis functions). By combining these aspects, the TempUnit should perform better than wavelet transform or FIR.

The limits of the current version of the TempUnit, first, concern the user-defined period of the basis function. Self-optimization of the duration of the basis function is foreseeable, and need to be linked to the inherent frequency content of the signal, as observed in the audio example. Second, the number of basis functions used for decomposition is also currently user-defined. However, there is an analytical solution to this problem (not shown) that will allow for an automatic solution in a future version of the TempUnit. Third, the TempUnit always needs a binary input, also in cases where no such signal is available. However, as illustrated by the audio example, this turned out to be an advantage in terms of coding the output in a very sparse manner.

In the following, we compare the characteristics of the TempUnit with those of a feed-forward TDMLP. Since TDMLPs are equivalent to FIRNNs, the comparison also applies implicitly to the latter. In comparison to a TDMLP, the TempUnit showed a much better performance: in average, the TempUnit was 2.3 times better. In addition, the TempUnit has many advantages over that of a TDMLP:

a) Whereas the TDMLP builds on a space-time conversion, i.e. each input unit represents a time step, the TempUnit has a "built-in" temporal operation.

b) This temporal mechanism is inspired by the temporal summation observed in biological neurons (e.g. [38]) and requires (in the present formulation) discrete time steps.

c) TDMLPs and recurrent networks do not provide explicit transfer functions. It is therefore difficult to interpret and formalize the network activity. The TempUnit in contrast provides an explicit basis function.

d) TDMLPs and reccurent NNs have relatively complex learning rules such as temporal backpropagation, whereas the TempUnit uses a simple learning rule based on a system of linear equations.

In general, there is no clear heuristic to optimize the architecture for TDMLPs or recurrent NNs. In contrast, there is a clear relationship between the TempUnit architecture and the characteristics of the signal. In particular, the TempUnit can be mapped to the theory of graphs. This will be explored in the future. Since supervised learning in TempUnit is based on linear system equations, it is not excluded that also analog inputs can be mapped to analog outputs. This would give the TempUnit model many more possible applications than shown here.

Acknowledgment

We are grateful to SN Baker, A Jackson and RN Lemon for having generously provided the experimental data obtained in the Sobell Department of Motor Neuroscience and Movement Disorders, Institute of Neurology, UCL, London, UK.

References

[1] Ballard, "Cortical connections and parallel processing," *Behavioral and Brain Sciences*, vol. 9, pp. 67–120, 1986.

[2] P. Baldi and W. Heiligenberg, "How sensory maps could enhance resolution through ordered arrangements of broadly tuned receivers.," *Biological Cybernetics*, vol. 59, pp. 313–318, 1988.

[3] H. Seung and H. Sompolinsky, "Simple models for reading neuronal population codes.," *Proceedings of the National Academy of Science USA*, vol. 90, pp. 10749–10753, 1993.

[4] E. Salinas and L. Abbott, "Transfer of coded information from sensory to motor networks," *Journal of Neuroscience*, vol. 15, pp. 6461–6474, 1995.

[5] E. Salinas and L. Abbott, "Vector reconstruction from .ring rates.," *Journal of Computational Neuroscience*,, vol. 1, pp. 89–107, 1994.

[6] H. Snippe, "Parameter extraction frompopulation codes:A critical assessment.," *Neural Computation*, vol. 8, pp. 511–529, 1996.

[7] A. Pouget, K. Zhang, S. Deneve, and P. Latham, "Statistically efficient estimation using population coding.," *Neural Computation*, vol. 10, pp. 373–401., 1998.

[8] K. Zhang, I. Ginzburg, B. McNaughton, and T. Sejnowski, "Interpreting neuronal population activity by reconstruction: Unified framework with application to hippocampal place cells.," *Journal of Neurophysiology*, vol. 79, pp. 1017–1044, 1998.

[9] P. Baraduc and E. Guigon, "Population computation of vectorial transformations.," *Neural Computation*, vol. 14, pp. 845–871, 2002.

[10] A. Georgopoulos, J. Lurito, M. Petrides, A. Schwartz, and J. Massey, "Mental rotation of the neuronal population vector.," *Science*, vol. 243, pp. 234–236, 1989.

[11] M. Wilson and B. McNaughton, "Dynamics of the hippocampal ensemble code for space.," *Science*, vol. 261, pp. 1055–1058, 1993.

[12] M. A. Maier, K. M. Bennett, M. C. Hepp-Reymonf, and R. N. Lemon, "Contribution of the monkey corticomotoneuronal system to the control of force in precision grip," *J. Neurophysiol.*, vol. 69, pp. 772-785, 1993.

[13] P. D. Cheney and E. E. Fetz, "Functional classes of primate corticomotoneuronal cells and their relation to actie force," *J. Neurophysiol.*, vol. 44, pp. 773-791, 1980.

[14] E. Guigon and P. Baraduc, "A neural model of perceptual-motor alignment.," *Journal of Cognitive Neuroscience*, vol. 14, pp. 538–549, 2002.

[15] P. Baraduc, E. Guigon, and Y. Burnod, "Recoding arm position to learn visuomotor transformations.," *Cerebral Cortex*, vol. 11, pp. 906–917, 2001.

[16] T. Poggio and E. Bizzi, "Generalization in vision and motor control.," *Nature*, vol. 431, pp. 768-74, 2004.

[17] D. Kleinfeld, " Sequential state generation by model neural networks.," *Proc. Natl Acad. Sci.*, vol. 83, pp. 9469-9473, 1986.

[18] H. Sompolinsky and I. Kanter, " Temporal association in asymmetric neural networks.," *Phys. Rev. Lett.*, vol. 57, pp. 2861-2864, 1986.

[19] A. V. M. Herz, "Global analysis of parallel analog networks with retarded feedback," *Phys. Rev.*, vol. A, pp. 1415-1418, 1991.

[20] A. Waibel, T. Hanazaqa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks.," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 37, pp. 328-339, 1989.

[21] E. Wan, "Temporal backpropagation for FIR neural networks," presented at International Joint Conference on Neural Networks, San Diego, 1990.

[22] E. Wan, "Time series prediction using a neural network with distributed time delays," presented at Proceedings of the NATO Advanced Research Workshop on Time Series Prediction and Analysis, Santa Fe, New Mexico, 1993.

[23] T. J. Cholewo and J. M. Zurada, "Exact Hessian Calculation in Feedforward FIR Neural Networks," presented at Proceedings of the IEEE International Joint Conference on Neural Networks, Anchorage, Alaska, USA, 1998.

[24] M. Abeles, *Corticonics: Neural Circuits of the Cerebral Cortex.* Cambridge: Cambridge University Press., 1991.

[25] M. Abeles, H. Bergman, I. Gat, I. Meilijson, E. Seidenmann, N. Tishby, and E. Vaadia, "Cortical Activity Flips among quasi-Stationary States.," *Proc. Natl. Acad. Sci.*, vol. 92, pp. 8616-8620, 1995.

[26] C. M. Gray, P. König, A. K. Engel, and W. Singer, "Oscillatory responses in cat visual cortex exhibit inter-columnar

synchronization which reflects global stimulus properties.," *Nature*, vol. 388, pp. 334-337, 1989.

[27] W. Singer, "Time as Coding Space in Neocortical Processing: A Hypothesis," in *Temporal Coding in the Brain*, G. e. a. Buzsaki, Ed. Berlin Heidelberg: Springer Verlag, 1994.

[28] M. Abeles and G. Gerstein, "Detecting Spatiotemporal Firing Patterns among Simultaneously Recorded Single Neurons," *J. Neurophysiol.*, vol. 60, pp. 909-924, 1988.

[29] M. Abeles, Y. Prut, H. Bergman, and E. Vaadia, "Synchronization in Neural Transmission and its Importance for Information Processing," *Prog. Brain. Res.*, vol. 102, pp. 395-404, 1994.

[30] Y. Prut, E. Vaadia, H. Bergman, I. Haalman, H. Slovin, and M. Abeles, "Spatiotemporal Structure of Cortical Activity: Properties and Behavioral Relevance.," *Journal of Neurophysiology*, vol. 79, pp. 2857-2874, 1998.

[31] Z. F. Mainen and T. J. Sejnowski, "Reliability of spike timing in neocortical neurons," *Science*, vol. 268, pp. 1503–1506, 1995.

[32] M. Flanders and U. Hermann, "Two components of Muscle Activation: Scaling with the speed of arm movement.," *Journal of Neurophysiology*, vol. 67, pp. 931-943, 1992.

[33] O. F. Manette and M. A. Maier, "Temporal processing in primate motor control: relation between cortical and EMG activity.," *IEEE Trans Neural Netw.*, vol. 15, pp. 1260-7, 2004.

[34] E. E. Fetz, S. I. Perlmutter, M. A. Maier, D. Flament, and P. A. Fortier, "Response patterns and postspike effects of premotor neurons in cervical spinal cord of behaving monkeys," *Can. J. Physiol. Pharmacol.*, vol. 74, pp. 531-546, 1996.

[35] E. E. Fetz and P. D. Cheney, "Post-spike facilitation of forelimb muscle activity by primate corticomotoneuronal cells," *J. Neurophysiol.*, vol. 44, pp. 751-772, 1980.

[36] R. N. Lemon, G. W. Mantel, and R. B. Muir, "Corticospinal facilitation of hand muscles during voluntary movement in the conscious monkey," *J. Physiol.*, vol. 381, pp. 497-527, 1986.

[37] S. N. Baker, R. Spinks, A. Jackson, and R. N. Lemon, "Synchronization in monkey motor cortex during a precision grip task. I. Task-dependent modulation in single-unit synchrony," *J. Neurophysiol.*, vol. 85, pp. 869-885, 2001.

[38] F. Rieke, D. Warland, R. De Ruyter Van Steveninck, and B. W., "Characterizing the neural response," in *Spikes*. Cambridge, MA: MIT Press, 1996.