Real-Time Musical Beat Induction with Spiking Neural Networks

Douglas S. Eck

Technical Report No. IDSIA-22-02

October 2002

IDSIA / USI-SUPSI
Instituto Dalle Molle di studi sull' intelligenza artificiale
Galleria 2
CH-6900 Manno, Switzerland

Real-Time Musical Beat Induction with Spiking Neural Networks

Douglas S. Eck

October 2002

Abstract

Beat induction is best described by analogy to the activities of hand clapping or foot tapping, and involves finding important metrical components in an auditory signal, usually music. Though beat induction is intuitively easy to understand it is difficult to define and still more difficult to perform automatically. We will present a model of beat induction that uses a spiking neural network as the underlying synchronization mechanism. This approach has some advantages over existing methods; it runs online, responds at many levels in the metrical hierarchy, and produces good results on performed music (Beatles piano performances encoded as MIDI).In this paper the model is described in some detail and simulation results are discussed.

1 Introduction

The term *beats* refers to sounds that are perceived as being equally spaced in time. *Downbeats* are particularly salient beats that usually occur at a comfortable tapping rate. When you tap your feet to the radio you are finding downbeats, a skill called *beat induction*. Beat acts as a unifying force, lending music the feeling of movement by providing a structure for the prediction of important musical events. Parncutt (1994) aptly wrote: “Imagine that you are walking down a quiet city alley and enter a jazz club. The door opens and suddenly you hear the music. In just a second or two you have a strong impression of the “feel” of the music—in particular the way it “swings”—its “beat” or “pulse.” (pp 409–410)” Though beat induction is relatively simple for people to perform—most of us would not find it difficult to tap our feet in Parncutt’s jazz club—it remains an open problem to build a convincing computational model (Eck, 2001; Large and Kolen, 1994; Cemgil et al., 2000).

One potential answer to this challenge lies in spiking neural networks (SNNs). SNNs are neural networks that explicitly model the temporal dynamics of neural firing. This approach differs from traditional feed-forward and recurrent neural networks where individual spikes are averaged into a single activation term. Though more difficult to simulate, SNNs benefit from an additional computational dimension. Namely, spike rate provides one dimension for encoding whatever activation encodes in traditional networks while spike synchrony provides a second dimension for encoding something different. Details are out of the scope of this paper; see Maass and Bishop (1998) for one of many good overviews.

One reason for the popularity of SNNs in neural network research is the belief that synchronous neural firing may help the brain bind multiple sensory streams into coherent representations. This hotly-debated “Temporal Binding Hypothesis” was first posed by Von der Malsburg (1981). Singer and Gray (1995) provided some of the first biological evidence in the visual cortex of a cat. See Shadlen and Movshon (1999) for an overview. A recurring observation in this research is that spiking neurons (real and simulated) are extremely robust, fast synchronizers even when coupled in very large groups (Somers and Kopell, 1993, 1995). Because fast synchronization with a temporal signal is at the heart of music perception and action, the question is raised of whether spiking neurons could perform beat induction.

To address this question we designed a model of beat induction that uses an SNN to synchronize with music. Input is presented to the network as voltage spikes obtained from a MIDI representation of music, either from a MIDI file or in real time from a MIDI musical instrument. Audio signals could also be processed directly using a method similar to Scheirer (1998); however this has not yet been tried. Neurons in the SNN are initialized with a range of frequencies suitable for rhythm (.5Hz to 5Hz). When exposed to a musical signal, clusters of neurons begin to fire in synchrony with periodic events in the signal. In many cases these clusters gravitate to metrically important events, including downbeats. When spike onsets are transformed into musical events, a listener can hear the network "drumming" along with a song.

The model was tested on a dataset of Beatles piano performances collected by the Music, Mind, Machine Group, NICI, University of Nijmegen (see Cemgil et al. (2000) or www.nici.kun.nl/mmm for more details). In general performance was good with the model finding downbeats reliably and quickly for good performances. With good parameter settings the model also failed gracefully on poor performances; unfortunately those settings were difficult to optimize such that the model worked on a wide range of performances. Because the model can be simulated quickly on modern PC hardware, it is possible to interact in real time using a standard MIDI keyboard. This yields results that are always interesting and sometimes even pleasing.

The remainder of the paper is as follows. Section 2 briefly describes previous attempts at beat induction. Section 3 introduces the model and Section 4 describes simulation results.

2 Background

When a neuron is fed a small amount of constant voltage it spikes with a regular frequency; that is, under some conditions a neuron oscillates. This indicates a strong link between SNNs and the nonlinear dynamics of physical and electrical oscillators. For this reason the SNN beat tracker can be placed among the family of oscillator beat tracking models that have been in the literature for at least fifteen years. Important contributions from this family are described briefly in this section. See also Eck (2002) and the introduction of Large and Jones (1999).

Dannenberg (Dannenberg, 1984; Allen and Dannenberg, 1984) used an oscillator to match performances to a score by finding downbeats in patterns. His model tracked acceleration and deceleration by modifying oscillator period in response to changes in pattern rate. Torras (1985) used firing threshold adaptation in limit oscillators similar to those commonly used in SNNs; the task was to find temporal regularities in simple rhythms. Miller et al. (1992) used a coupled one-dimensional network of oscillators (BEAT-NET) to resonate with rhythmical patterns.

These are early examples of explicit oscillator models. There are also examples where nonlinear oscillation is a model component, but is not a primary part of the system. For example, Todd et al. (1999) incorporated an oscillator model of musculoskeletal movement in a system that synchronizes body movements with temporal regularities in an input signal.

Similar oscillator models by McAuley (McAuley, 1994) and Large (Large and Kolen, 1994; Large and Jones, 1999) are successful at finding downbeats in patterns even when non-stationary noise (e.g. acceleration) is present in the patterns. McAuley used the term *adaptive oscillator* to describe a limit cycle oscillator that entrains both its phase and its period to recurring events in a temporal signal. The McAuley oscillator entrains to a rhythmical pulse train by discretely resetting its phase to zero when a pulse is sufficiently strong. This phase resetting is governed by a cosine-shaped function that is centered around the zero phase of the oscillator. By tightening the shape of this function, the system has the ability to focus on a particular periodic component of the signal, ignoring all others. The McAuley adaptive oscillator also attempts to match its period to periodic components in the signal. This is achieved by a function that slightly slows the oscillator when phase resetting consistently occurs early and slightly accelerates the oscillator when phase resetting consistently occurs late.

Large proposed a nonlinear limit cycle oscillator that entrains its phase to a rhythmical input by means of gradient descent. A smooth function that crosses zero at phase zero is used to establish a phase-based attractor. That function is minimized with respect to the difference between oscillator phase and the occurrence of input events, resulting in an oscillator that continually aligns its zero phase with that of events in the input. The Large oscillator has a second variable that modifies the slope and width of the gradient descent function such that the oscillator can sharpen its receptive field, allowing it to lock onto specific periodic components in the signal. Large and Kolen (1994) show that such an oscillator can form the basis of small connectionist networks that find salient events at multiple levels of the metrical hierarchy. Large and Jones (1999) extended these findings to a set of psychological experiments.

Non-oscillator beat trackers have also had success. However, because they are less closely related to the SNN model, only a very brief treatment is provided here; see Desain and Honing (1999) for one overview. Longuet-Higgins and Lee (1982) predicted downbeats by using a set of rules to minimize syncopation. Povel and Essens (1985) predicted downbeats by considering the relative fit of multiple clocks. Rosenthal (1992) assigned weighted scores to a set of hypothesis about the hierarchical structure of a musical performance. These scores matched the likelihood that a listener would prefer a particular interpretation. Desain and Honing (1991) used a settling neural network to adjust note onsets towards low integer ratios, performing quantization. Dixon (2001a) used a multi-agent algorithm to estimate tempo changes and downbeat onsets in digital audio or MIDI. Cemgil et al. (2000) formulated beat induction in a Baysean framework and use Kalman filters to find downbeats in a novel wavelet-based “Tempogram” representation.

3 An SNN model of beat induction

SNN architectures vary greatly and no common framework has been adopted (though the Spike Response Model (Gerstner, 1998) promises to serve this role). Our model is of a family of integrate-and-fire neural models which are simplifications of the original Hodgkin and Huxley (1952) neuron. We use the Fitzhugh-Nagumo neuron (Fitzhugh, 1961; Nagumo et al., 1962) which collapses the four variables of the Hodgkin-Huxley equations relating to electro-chemical processes in a neuron into two variables relating to the more abstract uptake and recovery of voltage. Under a wide range of parameter settings the Fitzhugh-Nagumo neuron exhibits the spiking dynamics of a real neuron: it gradually accrues voltage until it reaches a threshold; upon reaching that threshold it fires and quickly releases the energy. With constant and sufficient driving energy, this results in stable limit cycle oscillation. See Figure 1. The Fitzhugh-Nagumo equation for a single oscillator is

$$\begin{aligned} \dot{v} &= -v(v-\theta)(v-1) - w + \Omega \\ \dot{w} &= \epsilon(v - \gamma w) \end{aligned} \tag{1}$$

where v is voltage, w is recovery of voltage, θ is a threshold parameter fixed at 0.2 for all simulations, γ is a shunting parameter fixed at 2.5 for all simulations and Ω is any external voltage fed into the oscillator. A network is formed by adding voltage pulses from other oscillators into the Ω term. Voltage peaks are used to generate musical events (beats).

3.1 Coupling details

When an neuron fires, it sends a discrete pulse to other neurons via a series of weighted connections. In current experiments, all neurons were fully connected with uniform weak coupling strengths. Other solutions exist, including sparse coupling (Eck, 2000) and learned coupling strengths (Eck, 1999).

The musical input is treated in a way similar to input from other neurons. Note onsets in the MIDI source are transformed into weak pulses that are added to the omega (Ω) term in Equation 1. In these simulations the relative amplitudes of note onsets are preserved. However it can be difficult for the network to handle music with alternating quiet and loud passages. In this case a moving average method could be used that would allow the network to adjust input amplitude online.

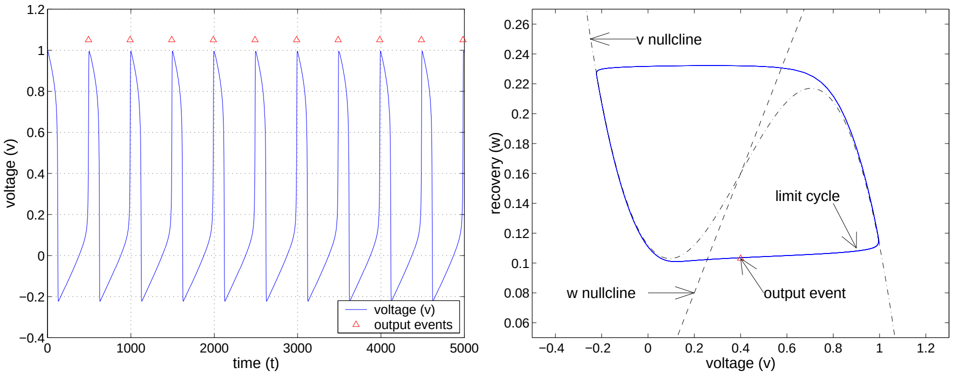

Figure 1: Time series (left) and phase portrait (right). Voltage peaks are used to generate discrete beats (triangles).

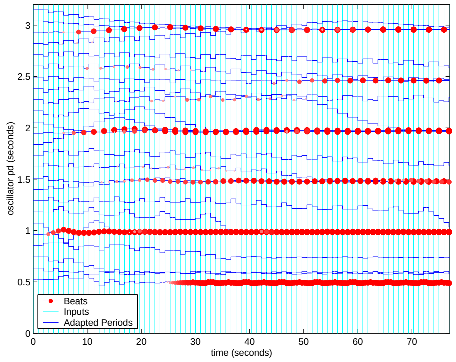

Figure 2: 25 neurons with a 60bpm metronome input. Neurons with high phase coherence are marked by the large circles. Note the migration of intrinsic periods towards harmonics of the metronome period.

3.2 Period adaptation

The neurons are started with intrinsic periods ranging over the rhythmical range (minimum .5Hz and maximum 5Hz, though some simulations use a narrower range). The neurons then modify their intrinsic period online in order to better track important periodic components. This is achieved by using a power-of-cosine based function that crosses zero when the neuron fires. Period is modified by adapting the epsilon (ϵ) term in Equation 1 as follows: $\epsilon(t) = \epsilon(t-1) + \delta f(x)$ where $f(x)$ is the period adaptation function and δ is a scalar. A similar method was used in, e.g., Large and Kolen (1994).

3.3 Tracking phase coherence

In a large network with many intrinsic periods, there will always be neurons which fail to track a given pattern. In general it sounds better to suppress those neurons so that their spurious spiking does not interfere with better-performing neurons. This is achieved by tracking the phase coherence (PC) of each neuron with the input signal. Whenever an input is encountered, the PC of neurons far from their firing point is lowered a small amount while the PC of neurons near their firing point is raised a relatively larger amount. See Somers and Kopell (1993) for a similar approach.

The PC value is used modulate the amplitude of neuron output pulses. It is also used to control the response of a neuron to input. Specifically, for neuron i, $\Omega_i = \Omega_i(1.0 - f(PC_i)\delta)$ where f is a bounded squashing function, usually the logistic sigmoid and δ is a scalar. Neurons with high

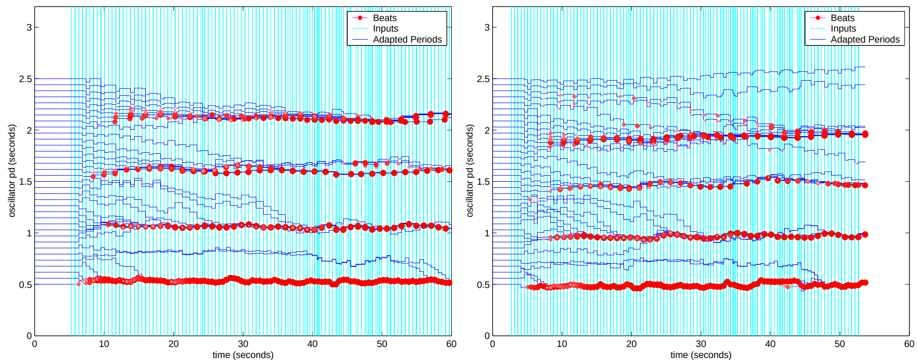

Figure 3: 35 neurons tracking *Michelle* played by a professional classical pianist at a medium tempo (left) and *Yesterday* played by a professional classical pianist at a fast tempo (right).

PC pay less attention to input pulses than to neurons with low PC. Similar strategies are used in many oscillator models, e.g. McAuley (1995).

For an illustration of both adaptation and phase coherence, see Figure 2. Here a network of 25 neurons receive input from a 60bpm metronome. Due to the effects of period adaptation, multiple neurons migrate to harmonics of the metronome period. As those neurons spike in synchrony with the input, their PC increases, as marked by the large circles.

4 Simulations

The model was tested on a dataset of Beatles piano performances collected by the Music, Mind, Machine Group, NICI, University of Nijmegen (see Cemgil et al. (2000) or www.nici.kun.nl/mmm for more details). Two songs were used, *Michelle* and *Yesterday* performed by 12 pianists with differing levels of experience (4 amateur classical, 4 professional classical, 4 professional jazz) at three tempo conditions (slow, medium and fast as judged by the performer), 3 times for each condition. This yielded 216 performances. These data were evaluated by Cemgil et al. (2000) and Dixon (2001b).

A network of 35 oscillators was used to process the pieces. Figure 3 shows a medium tempo performance of *Michelle* by a professional classical pianist. The network finds the quarter-note level (period near 0.5 sec) and half-note (period near 1.0 sec) level with high accuracy and the whole-note level (period near 2.0 sec) with less accuracy. Note that it also finds a spurious polyrhythm at period near 1.5. Suppressing these polyrhythms is a topic of current research.

To show that the model works with a wide range of performances, exactly the same network was used to process a fast tempo performance of *Yesterday* by the same professional classical pianist. See Figure 3. Network response was good but not perfect, with a strong spurious polyrhythm occurring at period 1.5.

4.1 Results

The performance of the model was promising, especially for medium tempo pieces and for performances without gross errors. However we do not offer a single percentage value of success for the dataset because of difficulties in unambiguously evaluating performance. In short, is difficult to know whether performance deviations are due to expressive timing or due to performance error. Thus when the model disagrees with the performance on the assignment of a beat it is difficult to know which is correct. This poses problems for the most obvious method of tracking error, looking for correlation between model beats and events in the input signal.

In our view the right way to evaluate the performacne of the model is to compare it to beat assignments made by a group of subjects. Unfortunately this data does not exits.

This raises the issue of whether the model should follow the performer (errors and all) or whether the model should ignore the performer when a gross error is made. The answer to this question depends in part on the task. If the task is to perform automatic score translation (i.e. turning performed music into sheet music) then the model must pay attention to and classify erroneous notes. The model should also work even for fairly large performance errors. If the task is to accompany a performer (i.e drumming along) then the model must at times ignore erroneous notes and place beats elsewhere. However, it is not so important that it track large performance errors; even human drummers have to restart if their playing partner makes a gross error.

5 Conclusions

We have proposed an SNN model of beat induction and described the model's behavior when run on some performed popular music. We have identified some shortcomings of the model: it works only in MIDI, is overly-sensitive to the amplitude of note onsets, and often finds spurious polyrhythms in a performance. Also, because the spiking rates of model neurons are slower than real neurons — musical rhythms have periods ranging from 5hz to .5hz — it cannot be thought of as neurologically plausible. However to our knowledge it is the only beat induction model that can extract beats at multiple metrical levels from performed music in real time. Clearly there is more work to do. However, we believe these preliminary results suggest that the model has promise as a perception-action layer in an intelligent musical device.

References

Allen, P. and Dannenberg, R. B. (1984). Tracking musical beats in real time. In *Proceedings of the International Computer Music Conference*, pages 140–143, International Computer Music Association.

Cemgil, A., Kappen, B., Desain, P., and Honing, H. (2000). On tempo tracking: Tempogram representation and kalman filtering. In Zannos, I., editor, *Proceedings of the 2000 International Computer Music Conference*, pages 352–355, Berlin. International Computer Music Association, The Berliner Kulturveranstaltungs GmbH.

Dannenberg, R. (1984). An on-line algorithm for real-time accompaniment. In *Proceedings of the 1984 International Computer Music Conference*, Computer Music Association.

Desain, P. and Honing, H. (1991). The quantization of musical time: A connectionist approach. In Todd, P. M. and Loy, D. G., editors, *Music and Connectionism*, pages 150–167. MIT Press, Cambridge, Mass.

Desain, P. and Honing, H. (1999). Computational models of beat induction: The rule-based approach. *Journal of New Music Research*, 28(1):29–42.

Dixon, S. E. (2001a). Automatic extraction of tempo and beat from expressive performances. *Journal of New Music Research*, 30(1):39–58.

Dixon, S. E. (2001b). An empirical comparison of tempo trackers. Eighth Brazilian Symposium on Computer Music, 31 July – 3 August 2001, Fortaleza, Brazil.

Eck, D. (1999). Learning simple metrical preferences in a network of Fitzhugh-Nagumo oscillators. In *The Proceedings of the Twenty-First Annual Conference of the Cognitive Science Society*, New Jersey. Lawrence Erlbaum Associates.

Eck, D. (2000). *Meter Through Synchrony: Processing Rhythmical Patterns with Relaxation Oscillators*. PhD thesis, Indiana University, Bloomington, IN, www.idsia.ch/~doug/-publications.html.

Eck, D. (2001). A positive-evidence model for rhythmical beat induction. *Journal of New Music Research*, 30(2):187–200.

Eck, D. (2002). Finding downbeats with a relaxation oscillator. *Psychological Research*, 66(1):18–25.

Fitzhugh, R. (1961). Impulses and physiological states in theoretical models of nerve membrane. *Biophysical Journal*, 1:455–466.

Gerstner, W. (1998). Spiking neurons. In Maass, W. and Bishop, C. M., editors, *Pulsed Neural Networks*, pages 3–53. The MIT Press.

Hodgkin, A. and Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117:500–544.

Large, E. W. and Jones, M. R. (1999). The dynamics of attending: How people track time-varying events. *Psychological Review*, 106(1):119–159.

Large, E. W. and Kolen, J. F. (1994). Resonance and the perception of musical meter. *Connection Science*, 6:177–208.

Longuet-Higgins, H. and Lee, C. (1982). The perception of musical rhythms. *Perception*, 11:115–128.

Maass, W. and Bishop, C. M. (1998). *Pulsed Neural Networks*. The MIT Press.

McAuley, J. (1995). *On the Perception of Time as Phase: Toward an Adaptive-Oscillator Model of Rhythm*. PhD thesis, Indiana University, Bloomington, IN.

McAuley, J. D. (1994). Finding metrical structure in time. In Mozer, M., Smolensky, P., Touretsky, D., Elman, J., and Weigend, A. S., editors, *Proceedings of the 1993 Connectionist Models Summer School*, pages 219–227, Hillsdale, NJ. Erlbaum.

Miller, B. O., Scarborough, D. L., and Jones, J. A. (1992). On the perception of meter. In Balaban, M., Ebcioğlu, K., and Laske, O., editors, *Understanding Music with AI: Perspectives on Music Cognition*, pages 429–447. MIT Press, Cambridge, Mass.

Nagumo, J., Arimoto, S., and Yoshizawa, S. (1962). An active pulse transmission line simulating nerve axon. *Proceeding IRE*, 50:2061–2070.

Parncutt, R. (1994). A perceptual model of pulse salience and metrical accent in musical rhythms. *Music Perception*, 11:409–464.

Povel, D. and Essens, P. (1985). Perception of temporal patterns. *Music Perception*, 2:411–440.

Rosenthal, D. (1992). Emulation of human rhythm perception. *Computer Music Journal*, 16(1):64–76.

Scheirer, E. (1998). Tempo and beat analysis of acoustic musical signals. *Journal of the Acoustical Society of America*, 103(1):588–601.

Shadlen, M. N. and Movshon, J. A. (1999). Synchrony unbound: a critical evaluation of the temporal binding hypothesis. *Neuron*, 24(1):67–77.

Singer, W. and Gray, C. (1995). Visual feature integration and the temporal correlation hypothesis. *Annual Review of Neuroscience*, 18:555–586.

Somers, D. and Kopell, N. (1993). Rapid synchronization through fast threshold modulation. *Biological Cybernetics*, 68:393–407.

Somers, D. and Kopell, N. (1995). Waves and synchrony in networks of oscillators of relaxation and non-relaxation type. *Physica D*, 89:169–183.

Todd, N., O'Boyle, D., and Lee, C. (1999). A sensory-motor theory of rhythm, time perception and beat induction. *J. New Music Research*, 28(1):5–28.

Torras, C. (1985). *Temporal-Pattern Learning in Neural Models*. Springer-Verlag, Berlin.

Von der Malsburg, C. (1981). The correlation theory of brain function. In Domany, E., van Hemmen, J., , and Schulten, K., editors, *Reprinted in Models of Neural Networks II (1994)*. Berlin: Springer. MPI Biophysical Chemistry, Internal Report 81 2.