POLYCHRONOUS WAVEFRONT COMPUTATIONS

EUGENE M. IZHIKEVICH*
The Neurosciences Institute,
10640 John Jay Hopkins Drive, San Diego, CA 92121, USA
Eugene.Izhikevich@nsi.edu

FRANK C. HOPPENSTEADT†
Courant Institute of Mathematical Sciences,
New York University, New York, NY, USA

There is great interest in methods for computing that *do not* involve digital machines. Many computational paradigms were inspired by brain research, such as Boolean neuronal logic [McCulloch & Pitts, 1943], the perceptron [Rosenblatt, 1958], attractor neural networks [Hopfield, 1982] and cellular neural nets [Chua & Yang, 1988]. All these paradigms abstract biological circuits to artificial neural networks, i.e. interconnected units (neurons) that perform computations based on the connections between the units (synapses). Here we present a novel computational framework based on polychronous wavefront dynamics. It is entirely different from an artificial neural network paradigm, rather it is based on temporal and spatial patterns of activity in pulse-propagating media and their interaction with transponders, which create pulses in response to receiving appropriate inputs, e.g. two coincident input pulses. A pulse propagates as a circular wave from its source to other transponders. Computations result from interactions between transponders, and they are encoded by the exact physical locations of transponders and by precise timings of pulses. We illustrate temporal pattern recognition, reverberating memory, temporal signal analysis and basic logical operations using polychronous wavefront computations. This work reveals novel principles for designing nanoscale computational devices.

Keywords: Networks; time delays; wavefront; analog computation.

1. Introduction

Polychronous wavefront computation is motivated by neuroscience: Synchrony of neuron firing is important in the brain, since two or more spikes arriving together elicit a stronger response in a post-synaptic neuron than if they arrive separately. Usually no account is made by researchers that there are axonal conduction delays, or if so, the delays are all assumed to be equal, so that synchronous firing results in synchronous arrival of spikes at postsynaptic targets. However, nerve impulses propagate along axons with finite velocity, and the axonal conduction delays depend on the distance between neurons, which can vary from neuron to neuron, often reaching tens of milliseconds [Swadlow, 1985, 1994]. Therefore, to ensure synchronous arrival, neurons must fire spikes at different moments of time with the temporal patterns being determined by conduction delays to the target. This phenomenon is

---

referred to as *polychronization* [Izhikevich, 2006], from Greek *poly* for *many* and *cronos* for *time* or *clock*. This phenomenon is illustrated in Fig. 1(a) where neurons *a* and *b* have different conduction delays (20 and 5 ms, respectively) to the target neuron *c*, reflecting different distances to *c*. If *a* and *b* fire together, the spikes they generate will arrive at *c* at different moments of time, see Fig. 1(b). A stronger input to *c* occurs if *b* fires 15 ms after *a*. A different temporal pattern might affect neuron *d* (if *a* fires 15 ms after *b*), etc.

The effect of coincidental input spikes to a biological neuron depends on many factors, such as the strength of synaptic conductances, the location of synapses on the dendritic tree, postsynaptic voltage-gated currents, neuromodulation and the history of neuronal activity. Taking into account all these factors is important for building detailed models of the brain [Izhikevich & Edelman, 2008]. However, our interest here is in more general kinds of transponders than neurons and more general kinds of media than networks of axons.

1.1. *Polychronous wavefront computations*

Our abstraction of these observations begins with a medium that can support the propagation of activity and in which there are (i) transponders whose sensitivities are tuned to fire a pulse in response to coincident arrival of two or more pulses, and (ii) the pulses they generate propagate outward from the source as *circular waves* through the medium.

The waves could pass through each other like deformation waves in an elastic medium or ripples on the surface of water (reaction–diffusion waves would annihilate each other upon collision and hence could not be used here).

One can visualize two ripples in still water where two pebbles have been tossed. The intersections of the ripples trace a parabola. A buoy located so as to lie on this parabola of intersection will experience a ripple of double amplitude compared to that were it not on the parabola. In response to a double ripple perturbation, the buoy can generate its own ripple, which can go on to interact with other transponders.

The wave-propagating medium corresponds to an axonal web in the brain with the exception that it is homogeneous and isotropic, i.e. waves can propagate in all directions, passing through each other, and affecting all transponders in the same way (no artificial synaptic weights). Since the propagation speed is finite, there result propagation time delays proportional to the distance between transponders. To fire or not to fire depends exclusively on the position of transponders and on the timing of input waves. This framework is simple enough to be implemented in physical media, yet it can perform nontrivial computations, as we show in this article.

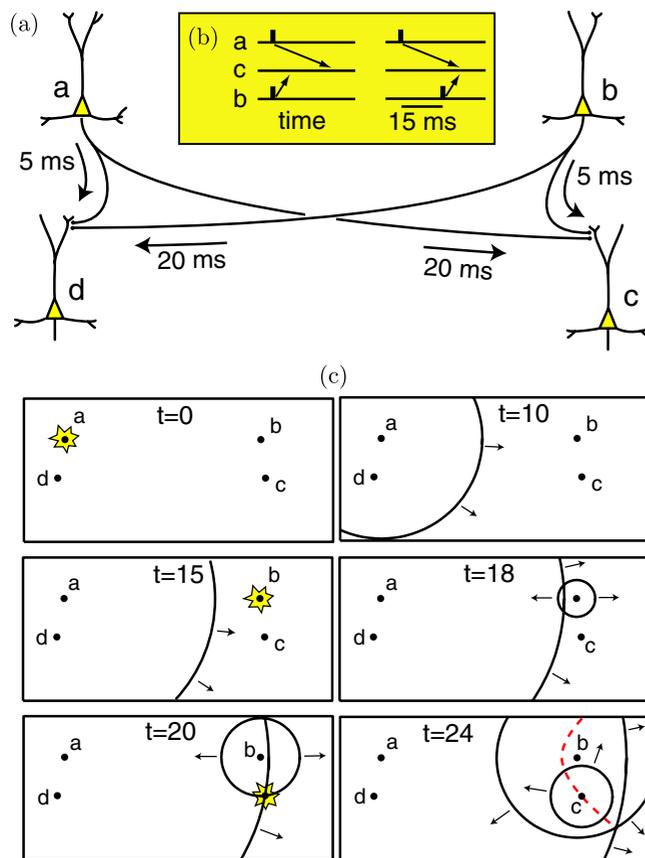We illustrate the main idea of polychronous wavefront computation in Fig. 1(c) using snapshots

Fig. 1. (a) Neurons *a* and *b* project to neuron *c* with axonal conduction delays of 20 ms and 5 ms, respectively. (b) If neurons *a* and *b* fire synchronously, their spikes arrive at neuron *c* at different times. If neuron *a* fires first and then neuron *b* fires 15 ms later, their spikes arrive at neuron *c* synchronously, evoking a stronger response in the neuron. (c) Transponders can generate waves of excitation that propagate through the medium. The transponders are located at such positions that a wave generated at transponder *a* (at $t = 0$), and one at transponder *b* (at $t = 15$) collide (at $t = 20$) at the location of transponder *c*, thereby exciting a response wave from it. The transponder *d* is unaffected. The dashed curve is the parabola traced by the intersections of waves emanating from *a* and from *b* with the inter-pulse interval of $t = 15$; any transponder on the parabola will fire in response to the pulses.

of two pulses from transponders a and b generated at times $t = 0$ and $t = 15$, respectively, and meeting at transponder c at $t = 20$. The pulse generated by a passes through d without any effect because d needs two pulses to fire. Similarly, the pulse from the transponder b passes through other transponders and through the pulse from a. The two pulses arrive coincidentally at the transponder c and evoke a pulse response there after a short latency period, which we use in our simulations as the unit of time. Obviously, the pulses from a and b fired at other times might excite other transponders.

Since the effect of initial pulses depends exclusively on their timing and on the relative positions of transponders in the system, we can program transponders by their placement to build circuits that can compute. Following are some examples that illustrate such polychronous wavefront computations.

2. Temporal Signal Processing

Polychronous wavefront computation can perform temporal signal analysis: Consider two transponders, a and b, that can be excited by external signals, e.g. from sensors or other transponders. Our goal here is to detect the instances when b is excited τ units of time after a, as in Fig. 2(a). If the two inputs correspond to auditory signals from two ears, then this architecture implements the Jeffress model of delay lines for binaural sound source localization in the azimuthal plane [Jeffress, 1948]. If pulses of a and b are triggered by the same signal, then the architecture can detect various inter-pulse intervals, and hence analyze the frequency content of the signal.

Pulses from a and b [Fig. 2(b)] trace a unique parabola depending on the inter-pulse interval. A few such parabolas are depicted in Fig. 2(c). To detect the interval τ, we put our "output" or "detector" transponder anywhere on the appropriate parabola. Since different inter-pulse intervals correspond to different parabolas in Fig. 2(c), other inter-pulse intervals can be detected by placing other transponders onto the corresponding parabolas. Each such transponder initiates its own wave, which could be used as the input to further transponders to detect complex inter-pulse patterns, e.g. triplets of pulses. This describes how polychronous wavefront computations can be programmed, and it lays a basis for performing Fourier computations, which we develop elsewhere.
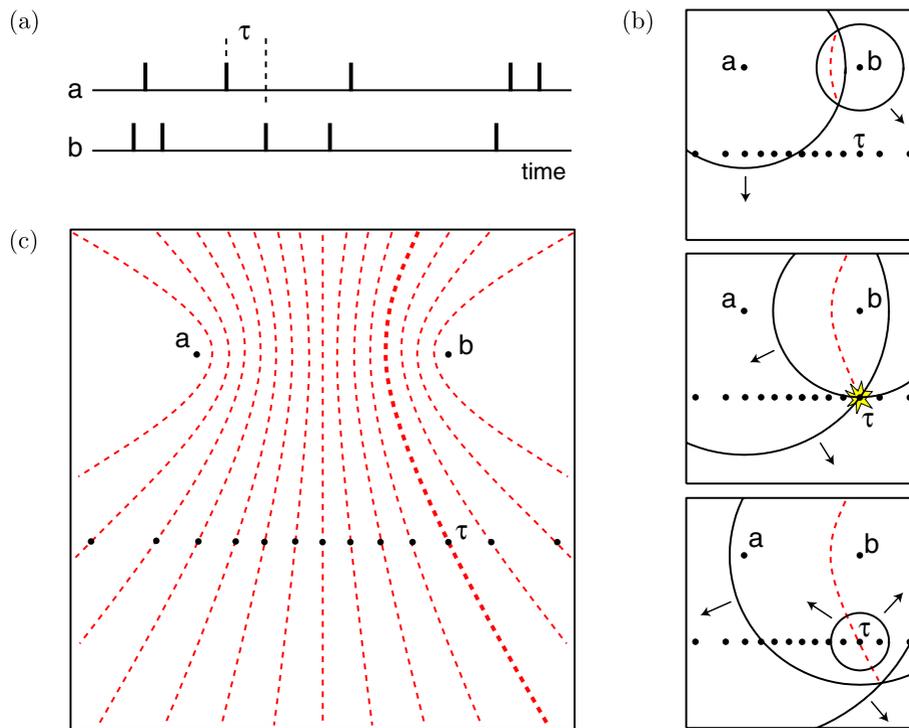

Fig. 2. (a) Detection of the inter-pulse interval τ in the input streams generated by transponders a and b. The interval elicits waves (shown in b) whose intersection traces a corresponding parabola in (c). The transponder labeled τ fires when the inter-pulse interval is τ.
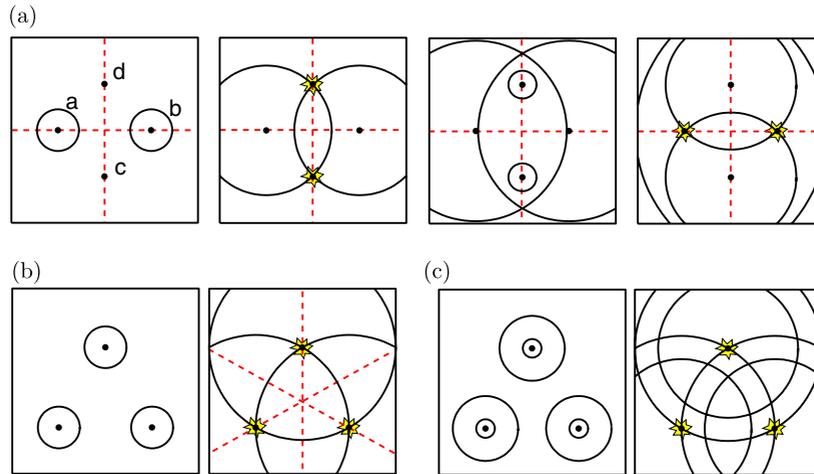
Fig. 3.   Reverberating memory units. Four (a) or three (b) equidistant transponders result in reverberating activity if triggered at the same time. (c) A reverberating doublet (two pulses) can encode and maintain any given real number as its inter-pulse interval. In these cases, the intersection parabolas are straight dashed lines.

3.   Reverberating Memory

Figure 3 depicts several configurations that can maintain stable reverberating activity, and hence serve as memory units. Consider transponders a and b in Fig. 3(a) firing at the same time. They emit waves whose intersections pass through c and d at the same time, thereby exciting these transponders and eliciting a new pair of waves, which return to a and b at the same time; the loop is closed and the activity is periodic. Interestingly, one can implement reverberating memory using only three transponders, as in Fig. 3(b).

One may think that the capacity of such a memory unit is only 1 bit, since at any given moment it either has a reverberating activity or it does not. However, the state of the unit is time-dependent, so we can encode information in the phase (i.e. timing) of excitation relative to some external timing signal. In this implementation, one could encode any real number into the phase of reverberating activity, theoretically making the memory capacity infinite. (In practice, it is bounded by the precision of the physical implementation of the system.) Figure 3(c) illustrates how periodic sequences of real numbers can be encoded into the reverberating memory unit: In this case, each transponder emits a doublet (two waves) with an inter-pulse interval corresponding to the "memorized" analog real number. The doublets reverberate, thereby maintaining the number.

4.   Logical Functions

Let us show that polychronous computations can be used to implement any general-purpose computations, e.g. via Boolean functions. For this, it suffices to show that we can implement Boolean NOT and OR (any other function is a superposition of these two; see [Whitehead & Russell, 1927]).

4.1.   *TRUE and FALSE states*

Before we proceed further, we need to define what states of excitable elements correspond to TRUE or FALSE signals. For example, a pulse may denote a TRUE signal and no pulse a FALSE signal, as in Fig. 4(a). This is not practical though, since a silent excitable element could be interpreted as constantly transmitting FALSE signals. Alternatively, a TRUE signal may correspond to two pulses, whereas a FALSE signal corresponds to one pulse [Fig. 4(b)]. Or, alternative to that, the inter-pulse interval may encode the truth value, as in Fig. 4(c). Finally, the timing of a pulse relative to some global signal may encode TRUE or FALSE values, as in Fig. 4(d). In this section, we use implementation in Fig. 4(e): The medium has two pacemaker excitable elements that trigger two waves periodically, called TRUE and FALSE waves, respectively. Any other element that is excited by the TRUE wave (and some other input) is said to emit TRUE signal, and any element excited by the FALSE wave (and some other input) is said to emit a FALSE signal. Because the waves pass via different elements at different times, the resulting pulses are not synchronous, but polychronous.

4.1.1.   *Boolean NOT*

Figure 5 depicts an implementation of the Boolean function NOT. It has two pacemaker (reference)
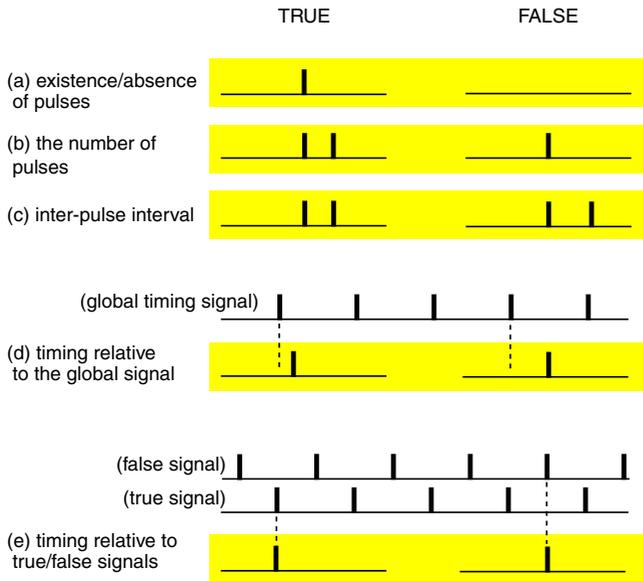
Fig. 4. Five different implementations of TRUE and FALSE states of an excitable element as described in the text.

excitable elements, the one on the bottom sends TRUE (green) reference waves and the one on the left sends FALSE (red) reference waves. All computations in the system are performed relative to those waves.

Suppose the input element in the middle is excited with the TRUE reference wave, as in Fig. 5(a). The output excitable element could be placed anywhere on the intersection of this wave and the FALSE reference wave, e.g. as in Fig. 5(b), so that it generates FALSE signal (i.e. it fires in response to the FALSE reference wave passing through it). Conversely, if the input is set to be excited by the FALSE reference wave, as in Fig. 5(d), the output excitable element may be anywhere on the intersection of this wave and the TRUE reference wave, e.g. as in Fig. 5(d), so that it generates a TRUE signal — it fires when the TRUE wave passes through it. That is, the excitable element must be on the intersection of the red dashed parabola in Fig. 5(c) and green dashed parabola in Fig. 5(f).

4.2. *Boolean OR*

Figure 6 depicts an implementation of the Boolean function OR. There are two input excitable elements marked by a and b in Fig. 6(a). Suppose they are activated with the TRUE references wave. There are three parabolas that interest us here: two that correspond to the intersection of the input waves with the next TRUE reference wave, marked by green dashed curves in Fig. 6(c) and the one that corresponds to the intersection of the input waves, marked by the black dashed curve in Fig. 6(c). The three parabolas intersect at a common point, marked by T in Fig. 6(c). We place our output excitable element at this point. The excitable element is activated by any two of the waves that converge here, so it would generate a TRUE signal if excitable element a is TRUE regardless of the value of b, excitable element b is TRUE regardless of the value of a, or both a and b are TRUE.
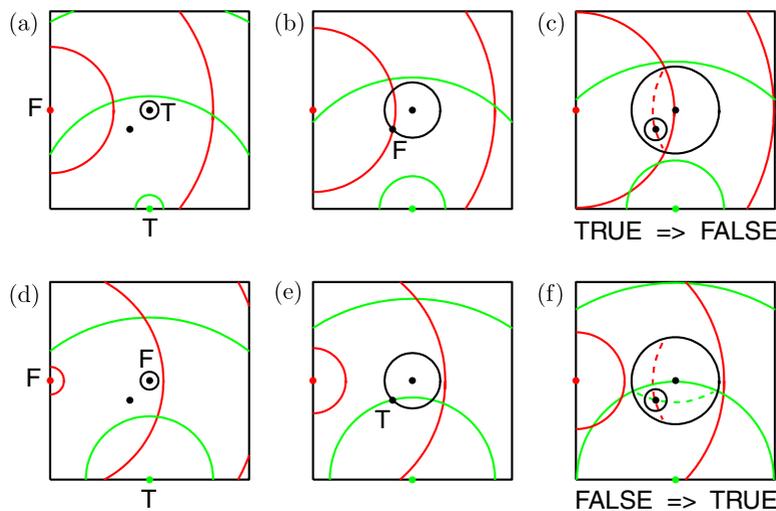

Fig. 5. Implementation of the Boolean function NOT. TRUE waves are depicted in green and FALSE waves are in red. The color is used only for the convenience of the reader, but it does not affect in any way the property of each wave. The black dots indicate carefully placed excitable elements as described in the text.
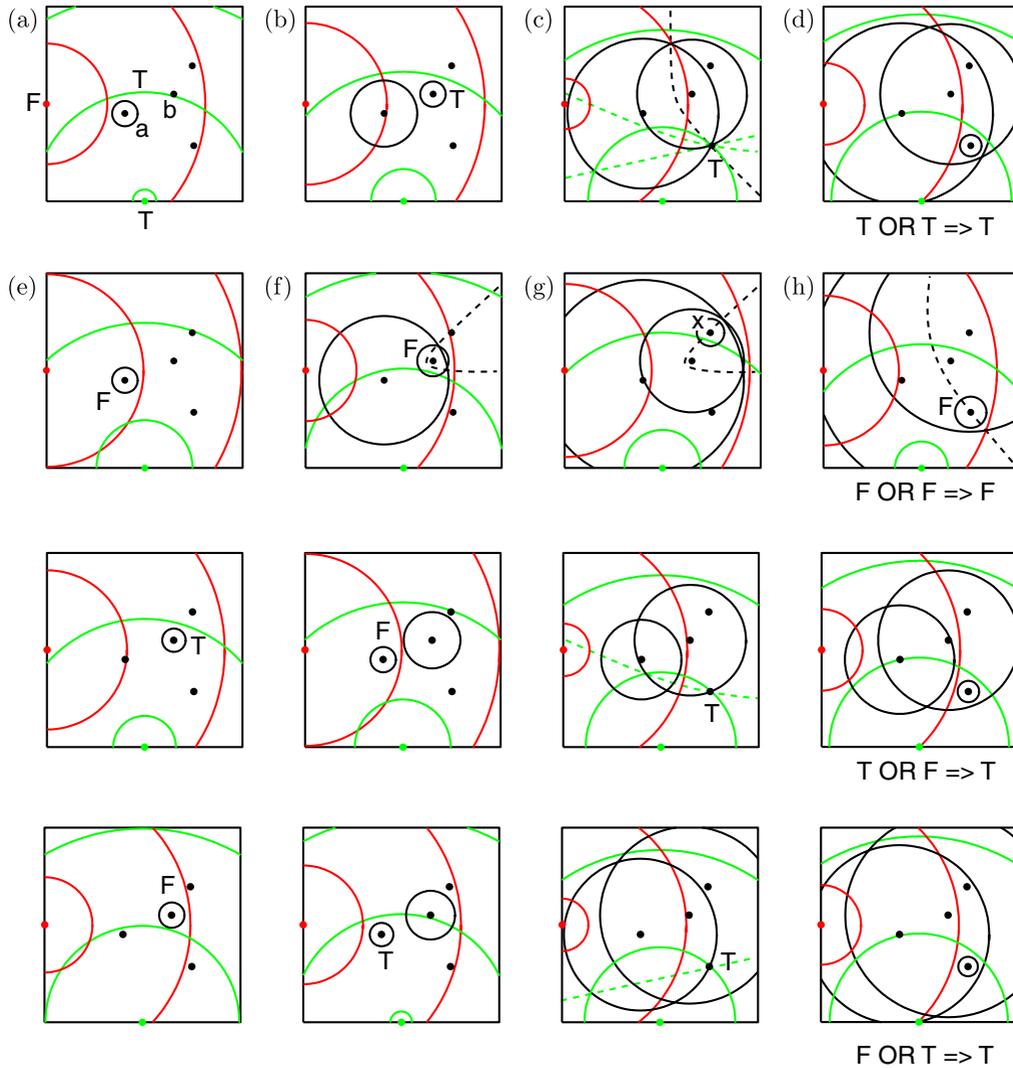
Fig. 6.   Implementation of the Boolean function OR. The black dots indicate carefully placed excitable elements as described in the text.

However, the output excitable element would be quiet if both a and b fire with the FALSE reference wave (as in Figs. 6(e)–6(h)). This is exactly what we need, because we want the output excitable element to fire with the FALSE reference wave in this case. For this, we put an auxiliary excitable element x on the parabola in Fig. 6(g) at such a location that the wave generated by x reaches the output excitable element at the same time as the FALSE reference wave passes through it.

The bottom half of Fig. 6 shows computations of the Boolean function OR when only one argument is TRUE. Apparently, the configuration is designed in such a way that one TRUE signal is enough to trigger the TRUE response regardless of the value of the other argument.

5. Discussion

We present here a theoretical paradigm for performing computations based on the difference of timing of pulses in a medium that supports isolated propagating waves, making possible polychronous wavefront computations. This mechanism was motivated by our studies of polychronization phenomena in spiking neural networks in the brain [Izhikevich, 2006]. This theory can be implemented in other physical systems having nothing to do with neuroscience; for example, arrays of Landau–Lifshitz–Gilbert models of nanosized spin torque oscillators [Slavin, 2006], cellular wave computers [Roska, 2005], MEMS oscillators, or phase-locked loops [Hoppensteadt & Lin, 2006], can be constructed that will produce comparable results of varying resolution to the theory described here.

Interestingly, chemical waves in reaction–diffusion systems would not work in this case, because excitation waves there collide and annihilate each other.

Any physical implementation, of course, would have certain limitations. For example, propagating waves and transponders would have a nonzero width; the speed of wave propagation may depend on its curvature (radius); propagating waves passing through each other may distort their phases; the waves may have decaying amplitudes; and to excite a transponder, two waves may need to have a certain (nonzero) angle of intersection. These last two features might actually be desirable in building large-scale circuits, since this would limit the scope of any computation to a local area.

The important difference between the polychronous wavefront architecture described here and other brain-inspired systems is that *it is not an artificial neural network*. Instead it is based on polychronous activity; it does not have synapses or any other specific connections that determine how one transponder affects another transponder. Here, the effect of each pulse depends exclusively on its timing relative to the other pulses and the locations of transponders, thereby emphasizing the polychronous nature of information processing.

Programming such a "polychronous wavefront computer" by placing transponders at certain locations is still an art rather than a science, although a graphical user interface is provided on http://www.izhikevich.com.

There are many open questions: How to use multilayer structures to extend the architecture in Fig. 2 to perform recognition of complex temporal patterns? Could activity-dependent adjustment of positions of transponders serve as a "learning rule"? How to use the memory units in Fig. 3 to implement simple arithmetic, such as addition of stored analog real numbers? Since the number of waves can grow exponentially, can this paradigm be used to solve notoriously hard, e.g. NP, problems? Are there any advantages in requiring that $n > 2$ coincident pulses are needed to excite a transponder? How can this paradigm be usefully extended into multidimensional spaces?

Acknowledgment

References

Chua, L. O. & Yang, L. [1988] "Cellular neural network: Theory," *IEEE Trans. Circuits Syst.* **35**, 1257–1272.

Hopfield, J. J. [1982] "Neural networks and physical systems with emergent collective computational abilities," *PNAS* **79**, 2554–2558.

Hoppensteadt, F. & Lin, K. [2006] "Propagating coalitions in networks of nonlinear oscillators," *Sci. Math. Japan* **64**, 779–784.

Izhikevich, E. M. [2006] "Polychronization: Computation with spikes," *Neural Comput.* **18**, 245–282.

Izhikevich, E. M. & Edelman, G. M. [2008] "Large-scale model of mammalian thalamocortical systems," *PNAS* **105**, 3593–3598.

Jeffress, L. A. [1948] "A place theory of sound localization," *J. Compar. Physiol. Psychol.* **41**, 35–39.

McCulloch, W. & Pitts, W. [1943] "A logical calculus of ideas immanent in nervous activity," *Bull. Math. Biophys.* **5**, 115–133.

Rosenblatt, F. [1958] "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.* **65**, 386–408.

Roska, T. [2005] "Cellular wave computers for brain-like spatial-temporal sensory computing," *IEEE Circuits Syst. Mag.* **19**, 5–19.

Slavin, A. [2006] "Theory of mutual phase-locking of spin-torque nanosized oscillators," *Phys. Rev. B* **74**, 104–401.

Swadlow, H. A. [1985] "Physiological properties of individual cerebral axons studied *in vivo* for as long as one year," *J. Neurophysiol.* **54**, 1346–1362.

Swadlow, H. A. [1994] "Efferent neurons and suspected interneurons in motor cortex of the awake rabbit: Axonal properties, sensory receptive fields, and subthreshold synaptic inputs," *J. Neurophysiol.* **71**, 437–453.

Whitehead, A. N. & Russell, B. [1927] *Principia Mathematica* (Cambridge University Press, New York).