Transient Responses in Dynamical Neural Models*

Michael Stiber

Department of Computer Science
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon
Hong Kong
email: stiber@cs.ust.hk

José P. Segundo

Department of Anatomy and Cell Biology
and Brain Research Institute
University of California
Los Angeles, California 90024
email: iaqfjps@mvs.oac.ucla.edu

Abstract

We consider the input/output behavior of a realistic dynamical neural model in comparison to those typically used in artificial neural networks. We have found that such models duplicate well those behaviors seen in living neurons, displaying a range of behaviors commonly seen in a wide variety of nonlinear dynamical systems. This is not captured well by weighted sum/monotonic transfer function models. An example of the consequences of nonlinear dynamics in neural responses is presented for monotonically changing input transients.

1 Dynamical Neural Networks

Though our knowledge of the computational capabilities of feedforward neural networks is now extensive, those which have internal states and dynamical behavior remain somewhat problematic. One reason for this is the *bifurcation behavior* of nonlinear dynamical systems, where small changes in a network parameter, such as a weight or net input to a unit, can switch the system between drastically different transfer functions. This has been the focus of recent work in learning in recurrent networks [1], but less is known about the computational implications for networks in which each unit is a nonlinear dynamical system in its own right.

Certainly, for realism's sake, one must consider individual units' dynamics to be significant for network function [2]. Biological nervous systems are networks of synaptically-coupled neurons, each producing sequences of short-lived voltage *spikes*, also called *action potentials* (APs). The input/output behavior of each neuron can be thought of as the transformation of trains of input spikes, via synapses, into trains of output spikes. This *synaptic coding* is the operational unit of nervous systems [3], and understanding it a necessary, though not sufficient, step towards understanding neural computation.

*This work was supported by the Hong Kong Research Grants Council (DAG92/93.EG18) and Trent H. Wells, Jr., Inc. ODEPACK software, developed at Lawrence Livermore Laboratories, and libf77 routines, copyright 1990 & 1991 by AT&T Bell Laboratories and Bellcore, were used.

The complexity of neuron input/output behavior has been known for quite some time [4], and has more recently been studied using techniques from the field of nonlinear dynamics [5, 6, 7].

2 Simulation and Analysis Methods

The analysis techniques and model used in these simulations is described in detail in the literature [5, 8, 3, 2, 9, 10, 11]; the following is a brief summary.

We used a physiologically-derived model of the lobster slowly adapting stretch receptor organ (SAO) [12]. This was modified to match the unperturbed behavior of the crayfish SAO [8, 9], on which similar experiments were performed [11]. This was used as the *driven* cell in a 2-cell network, connected to receive the outputs of a *driver* cell via an inhibitory synapse (the SAO is considered to contain a prototypical inhibitory synapse). The model has been shown previously to duplicate most of the behaviors of the SAO [10].

We analyzed the timing relationships between (presynaptic) input spike trains and (postsynaptic) output spike trains, each train assimilated to a point process [13]. The experimental methods are schematized in Figure 1. Monotonically changing *transients* were applied, where the instantaneous rate (A) increased or decreased; the intervals between presynaptic spikes (I_k, B) decreased or increased, respectively. The times of presynaptic and postsynaptic (C) events were recorded, yielding two sets of event times, s_k and t_i, where $k, i = 0, 1, 2, \ldots$. Each transient was followed by sufficient "dead time" (without any inputs applied) for the neuron to return to its unperturbed state.

Timing was analyzed in terms of certain intervals and cross intervals between events, $I_k = s_k - s_{k-1}$, $T_i = t_i - t_{i-1}$, and ϕ_i, the cross interval from t_i back to the most recent preceding presynaptic event (the latter called the *phase*) [5]. The example used in this paper was an accelerating ramp transient, with initial rate $1/I_b = 0.5$Hz, final rate $1/Ie = 16.67$Hz, and time *span* $S = 30$s.

Here we will illustrate the behavioral consequences of these transients with *pooled graphs* (Figure 1(E), in which we plot ongoing time since the start of a transient along the X axis and either I_k, T_i, or ϕ_i on the Y axis. These graphs are pooled because the points for all of the transients are

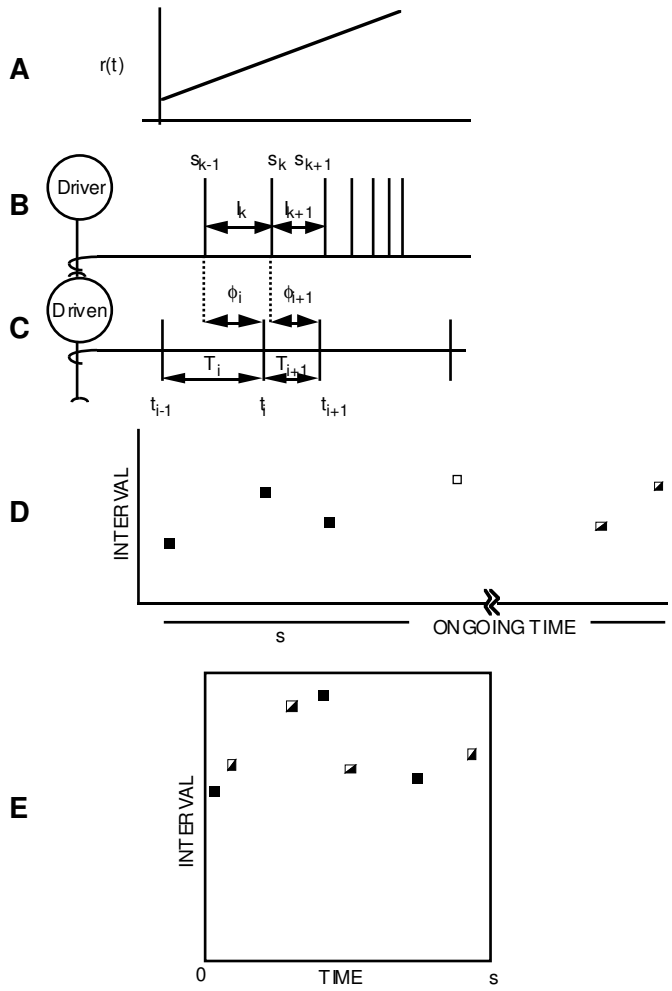

Figure 1: Summary of simulation and analysis methods. A monotonically changing rate control (A). Presynaptic neuron produced spikes with monotonically changing intervals, I_k (B). Postsynaptic spike times, t_i, were recorded (C). Intervals T_i and ϕ_i were computed (C), and plotted versus time (D). Intervals occurring during stimulation time S were plotted versus time since beginning of stimulation in pooled graphs (E).

superimposed; the X axis shows time since the beginning of *any* transient and a point may be from any transient, as illustrated in the figure by the differently shaded rectangles.

3 Results

As the accelerating transient sweeps from long intervals to short, one might naively expect that their inhibitory nature would be reflected in a decelerating trend in the model neuron's output. This is only correct in terms of the overall average trend that we see in the output intervals in Figure 2; any local region shows either an orderly *accelerating* trend or a disorderly sequence of outputs that is difficult to describe.

In the phase pooled graph in Figure 3, orderly interval regions are found to correspond to orderly phase regions; similarly for disorderly regions. Such orderly regions correspond to *alternations*, in which the neuron's outputs occur in a fixed temporal relationship to the inputs it receives. This is the nonstationary generalization of the stationary behavior called *locking*. Because of this fixed timing, the trend in the output matches that of the input — opposite of what one would expect naively from inhibitory input. Additionally, we can determine dynamical behavioral categories for the disorderly regions, too, including *quasiperiodic*, *intermittent*, and *chaotic*. This is a matter of ongoing work [14], which will also discuss the effects of the various input parameters (transient time span, waveform, and start and end rates) on the postsynaptic response.

4 Conclusions

When we move away from simplifying, nondynamical models of neurons, we find that their input/output behavior is complex and certainly is not adequately describable in terms of weighted sums and monotonic rate transfer functions, at least when considering the coding performed by a single

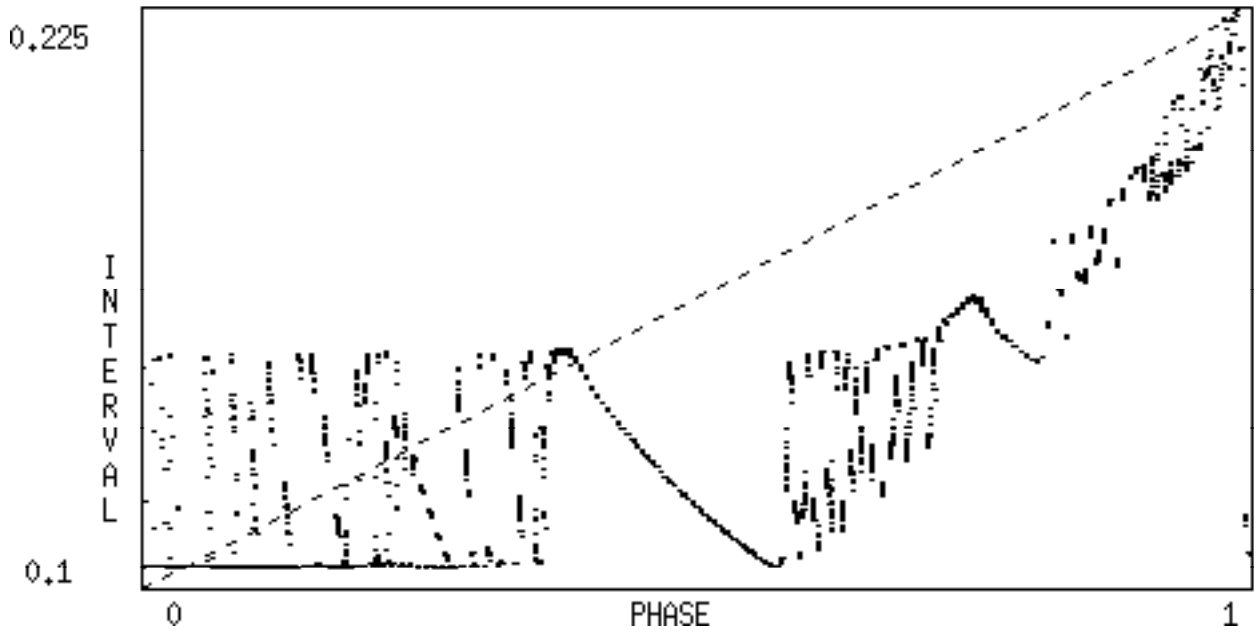

Figure 2: Pooled graph of output spike intervals, T_i, in response to an accelerating ramp transient. Note overall lengthening trend of intervals, significant negatively-sloped regions, and sudden bifurcations between different output patterns.

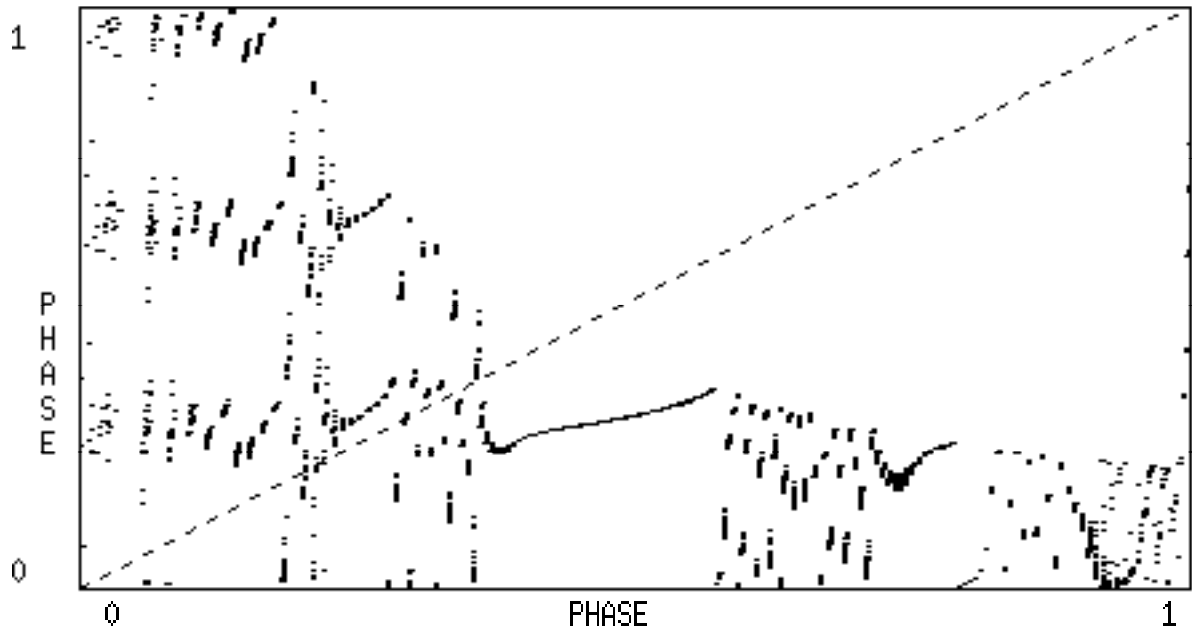

Figure 3: Pooled graph of output spike phases, ϕ_i, in response to an accelerating ramp transient. Note abrupt bifurcations between orderly and disordered patterns, and increasing compression of phases into a narrow range.

inhibitory synapse.

As in the SAO, we have seen that monotonic, smooth inhibitory input transients do *not* code into oppositely-sloped outputs, in spite of the inhibitory nature of the synapse. Nevertheless, the form of the pooled graphs indicates that each transient produced the same behaviors during the same time intervals. Otherwise, the superimposition of many transient outputs would not produce clearly identifiable indications of behaviors such as alternation, etc. This underscores the fact that this code is deterministic, a prerequisite for a computational structure.

The computational significance of these observations, and how they might apply to more complex input patterns, is currently being investigated. However, we can conclude that, if a monotonic coding is required, then connectivity in a network composed of realistic neural models will necessarily be quite complex. Contrastingly, the complex coding seen with simple connectivity seems to indicate that the computational complexity of even small networks may be significant, and that those concerned with building artificial neural networks may ignore the computational power of individual units or connections at their own risk.

References

[1] K. Doya, "Bifurcations of recurrent neural networks in gradient descent learning," *IEEE Transactions on Neural Networks*, submitted, 1993.

[2] M. Stiber and J. P. Segundo, "Dynamics of synaptic transfer in living and simulated neurons," in *Proc. IEEE International Conference on Neural Networks*, (San Francisco), pp. 75–80, March 1993.

[3] J. Segundo, J.-F. Vibert, M. Stiber, and S. Hanneton, "Synaptic coding of periodically modulated spike trains," in *Proc. IEEE International Conference on Neural Networks*, (San Francisco), pp. 58–63, March 1993.

[4] J. Segundo and D. Perkel, "The nerve cell as an analyzer of spike trains," in *The Interneuron: UCLA Forum in Medical Sciences* (M. Brazier, ed.), (Los Angeles), pp. 349–89, University of California Press, 1969.

[5] J. P. Segundo, E. Altshuler, M. Stiber, and A. Garfinkel, "Periodic inhibition of living pacemaker neurons: I. Locked, intermittent, messy, and hopping behaviors," *Int. J. Bifurcation and Chaos*, vol. 1, pp. 549–81, September 1991.

[6] J. P. Segundo, E. Altshuler, M. Stiber, and A. Garfinkel, "Periodic inhibition of living pacemaker neurons: II. Influences of driver rates and transients and of non-driven post-synaptic rates," *Int. J. Bifurcation and Chaos*, vol. 1, pp. 873–90, December 1991.

[7] J. Segundo, M. Stiber, E. Altshuler, and J.-F. Vibert, "Transients in the inhibitory driving of neurons and their post-synaptic consequences," *Neuroscience*, 1993. submitted.

[8] M. Stiber, *Dynamics of Synaptic Integration*. PhD thesis, University of California, Los Angeles, 1992.

[9] M. Stiber, J. P. Segundo, T. Nomura, S. Sato, S. Doi, J.-F. Vibert, and K. Pakdaman, "Biological computation: Neurons, networks, and nonlinear dynamics," in *Biological Neural Networks* (D. Tam, ed.), Norwood, NJ: Ablex, 1993. submitted, also Technical Report HKUST-CS93-8.

[10] M. Stiber, J. P. Segundo, E. Altshuler, and N. Jurisic, "Complex, complex enough, too complex? A comparison of dynamical neural models," *Neural Networks*, 1993. submitted, also Technical Report HKUST-CS93-11.

[11] J. Segundo, M. Stiber, E. Altshuler, and J.-F. Vibert, "Transients in the inhibitory driving of neurons and their post-synaptic consequences," *Neuroscience*, 1994. in press.

[12] Å. Edman, S. Gestrelius, and W. Grampp, "Analysis of gated membrane currents and mechanisms of firing control in the rapidly adapting lobster stretch receptor neurone," *J. Physiol.*, vol. 384, pp. 649–69, 1987.

[13] D. Cox and V. Isham, *Point Processes*. London: Chapman and Hall, 1980.

[14] M. Stiber, J. P. Segundo, and J.-F. Vibert, "Transient responses of a dynamical neuron simulation." in preparation, 1994.