Maximum likelihood decoding of moving stimuli using divisive normalization line attractor neural networks

Robert C. Wilson and Leif H. Finkel

Department of Bioengineering, University of Pennsylvania, Philadelphia, PA 19104, USA

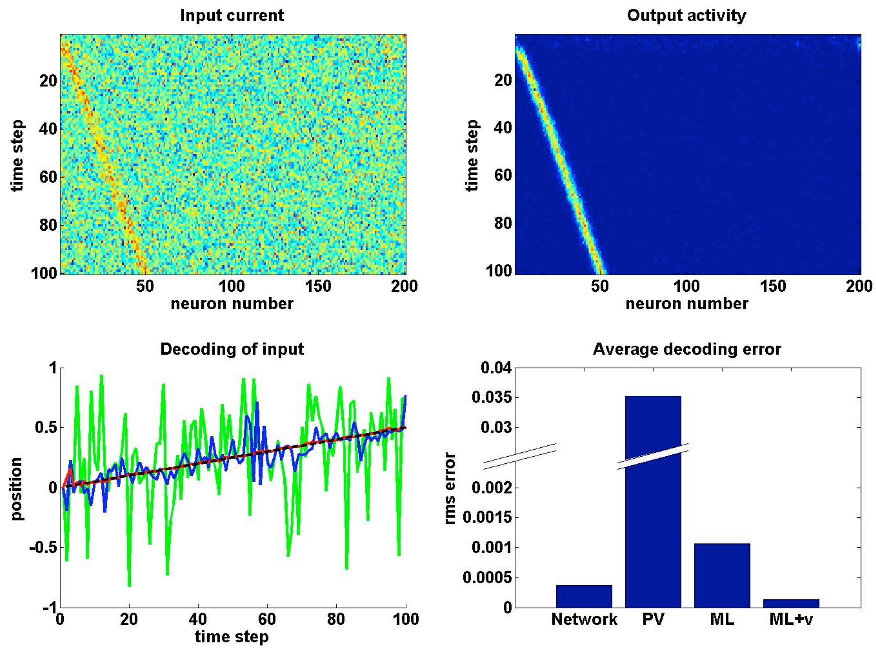
There is a growing body of evidence that the brain computes optimally. However, despite some excellent theoretical suggestions (e.g. Gold, J.I. and Shadlen, M.N., Nature, 404: 390-394, 2000; Deneve, S., NIPS. 13. 2004; Rao, R. Neural Computation, 16(1), 1-38, 2004; Zemel, R.S., Huys, Q.J.M., Natarajan, R and Dayan, P. NIPS 2004**)**, precisely how the brain achieves this feat is still an open question. A particularly promising approach was suggested by Deneve et al. (Deneve, S., Latham, P.E. and Pouget, A. Nature Neuroscience. 2(8):740-745. 1999) where the authors used a line attractor neural network to optimally decode a *static* stimulus.

In this work we extend the network of Deneve et al. to deal with moving stimuli. We do this by connecting up a line attractor network in such a way that the stable state of the system is a rotating bump of activity whose speed can be altered by changing the input to different parts of the network. In particular we use a double ring line attractor network similar to that suggested in Zhang 1996 (Zhang K., J Neurosci. 16:2112-2126, 1996) and Xie et al. 2002 (Xie, X., Hahnloser, R.H.R., and Seung, H. S. , Physical Review E 66, 041902, 2002), but with the difference that the activation rule implements divisive normalization. We use the divisive normalization rule as this is both biologically plausible (e.g. see Carradini, M., Heeger, D.J., and Movsham, J.A., J. Neurosci. 17(21), 8621-8644, 1997) and leads to robust line attractors.

The aim of our network is to decode the position of a moving stimulus (e.g. a moving dot) given a noisy input signal as a function of time. The network then receives three types of inputs: the speed of the stimulus, which could be provided by MT cells; the external input comprising a noisy hill of activity centered around the current position of the stimulus; and the recurrent inputs from the network itself.

The figure below demonstrates the basic properties of our model. The moving stimulus creates a noisy input current (shown in the top left panel) whose position is determined by the position of the stimulus. In the present case, the noise is created by simply adding uncorrelated Gaussian noise to the input current at each time step. This input is fed into the divisive normalization network and the output of this network is shown in the top right panel. Clearly this is a much cleaner signal and the position can easily be decoded using a population vector decoder. This decoded signal is shown in the bottom left panel. The green line represents population vector decoding of the input signal; the blue line is the decoding from a divisive normalization network, but with the speed set to zero – which is equivalent to the network used in Deneve et al.; the red line is the decoding using our network; and the dashed black line is the actual motion. Clearly our network significantly outperforms the other decoding methods.

The results of a more detailed empirical investigation are shown in the bottom right panel. In these experiments we compared the network decoding ('Network') to population vector ('PV') decoding; and to two versions of maximum likelihood decoding (ML and ML+v). The first, ML, is the optimal decoding of the stimulus position without taking into account any velocity information, i.e. this estimator makes no assumptions that the position of the stimulus at different time points will be correlated in any way. The second estimator, ML+v, assumes that the stimulus is moving at speed v and finds the optimal straight line path of speed v through the activity. This second version is the best possible decoding of the inputs that the network receives. In the figure we plot the rms error over 20 different trials each lasting 100 time steps for these different decoding schemes. These results clearly show that our network performs close to optimally and that the ability of the line attractor network to integrate information over time gives it an advantage over the uncorrelated, ML decoding scheme.



Input current
time step
neuron number
Output activity
time step
neuron number
Decoding of input
position
time step
Average decoding error
rms error
Network
PV
ML
ML+v