ReScience

[Re] A neural model of the saccade generator in the reticular formation

Mario Senden[1, 2], **Jannis Schuecker,**[3], **Jan Hahne,**[4], **Markus Diesmann,**[3, 5, 6], **and Rainer Goebel,**[1, 2, 7]

1 Department of Cognitive Neuroscience, Faculty of Psychology and Neuroscience, Maastricht University, 6201BC Maastricht, The Netherlands **2** Maastricht Brain Imaging Centre, Faculty of Psychology and Neuroscience, Maastricht University, P.O. Box 616, 6200 MD Maastricht, The Netherlands **3** Institute of Neuroscience and Medicine (INM-6) and Institute for Advanced Simulation (IAS-6) and JARA BRAIN Institute I, Jülich Research Centre, 52428 Jülich, Germany **4** School of Mathematics and Natural Sciences, Bergische Universit"at Wuppertal, Wuppertal, Germany **5** Department of Psychiatry, Psychotherapy and Psychosomatics, Medical Faculty, RWTH Aachen University, 52062 Aachen, Germany **6** Department of Physics, Faculty 1, RWTH Aachen University, 52062 Aachen, Germany **7** Department of Neuroimaging and Neuromodeling, Netherlands Institute for Neuroscience, an Institute of the Royal Netherlands Academy of Arts and Sciences (KNAW), 1105BA Amsterdam, The Netherlands

mario.senden@maastrichtuniversity.nl

Editor
Name Surname

Reviewers
Name Surname
Name Surname

Received Month, Day, 2017
Accepted Month, Day, 2017
Published Month, Day, 2017

Competing Interests:
The authors have declared that no competing interests exist.

Article repository

Code repository

A reference implementation of

→ *A neural model of the saccade generator in the reticular formation*, G. Gancarz, S. Grossberg, Neural Networks, 1159-1174, 1998

Introduction

We provide an implementation of the saccade generator (SG); a rate neuron model of the neural circuitry in the reticular formation proposed by Gancarz & Grossberg [2]. The same group has recently sucessfully embedded the SG into a larger model of the eye movement network [4] showcasing its compatible nature. This compatibility of the SG model might prove useful in the future for studying the interplay of neural (sub)systems of visuo-motor integration. It is thus of interest to implement the model in publicly available, widely used, and actively developed neural simulation frameworks such as NEST [3]. We show that the model translates well to the NEST framework as our implementation faithfully reproduces most simulation results reported in the original publication. Our code uses the Python interface [1] for legibility with both model and analysis scripts being implemented using Python 2.7.12.

Methods

The SG model described by Gancarz & Grossberg [2] consists of a horizontal and a vertical component each with two long-lead burst neurons (LLBNs), excitatory burst neurons (EBNs), inhibitory burst neurons (IBNs), and tonic neurons (TNs). Within each component, the two directions (left-right, down-up) interact antagonistically. Additionally, both components share a single omnipause neuron (OPN) which tonically inhibits each EBN as long as no saccade is being initiated. In implementing this model, we largely followed the descriptions provided in the original publication with a number of well-motivated exceptions. First, in the original description, neuron activations are bounded from below at zero implying their rectification at every step during numerical integration. Instead, we opted for passing all inputs received by a neuron through a rectified linear gain function prior to their summation. Second, we replaced the gain function given by equation A11 in the original publication

$$g(x) = \frac{x^4}{0.1^4 + x^4} \tag{1}$$

by

$$g(x) = \frac{1}{1 + e^{-40(x-0.1)}} \tag{2}$$

to prevent positive responses to negative net input. Otherwise equation 2 closely approximates the shape of equation 1 for $\mathrm{x} > 0$. Third, according to equation A12 in the original publication, horizontal eye position is given by $\theta = 260(\mathrm{TN_r} - 0.5)$. However, omitting rectification in our implementation, allowed us to set activation of TNs to 0 rather than 0.5 when the eye is at the center of its range. Finally, the original implementation uses the fourth order Runge–Kutta method for numerical integration. Instead, we used the Exponential Euler method which is standardly implemented in NEST 1.1.x.x for numerical integration of rate neurons [5].

In addition to these changes, the original model description has two features which cannot be straightforwardly translated to NEST. First, a nonlinear gain function is applied to a subset of inputs to EBNs and the OPN while a linear gain function is applied to their remaining inputs. Since NEST only applies a single gain function per neuron to each of its inputs, we opted for using a (rectified) linear gain function for EBNs and the OPN. In addition, we passed those inputs requiring an additional nonlinear gain function through an auxiliary unit instantaneously applying the desired nonlinearity before passing the result on to EBNs and the OPN. Second, constant input to a neuron was not hard-coded but rather provided by an appropriately weighted bias node.

In all simulations we used a time step of 0.001 ms and a time constant of 50 ms. It should be noted that rates of all neurons were initialized to zero and were thus not at resting equilibrium. In order to address this, the model was allowed to evolve for 50 ms, to reach equilibrium before applying any input. Furthermore, we always simulated the full model; i.e. both its horizontal and vertical components even if input was applied only to one of the two.

Results

All results from the original publication have been implemented. Our results accord very well with those reported by Gancarz & Grossberg [2]. Nevertheless, some inconsistencies exist and will be discussed.

The first simulation reported in Gancarz & Grossberg [2] showcases the evolution of activity for each neuron type in the horizontal SG to a constant input ($\mathrm{I} = 1$) applied to the left LLBN for 265 ms. The original publication does not report exact activation values observed for each neuron rendering a quantitative analysis of the accuracy of our replication impossible. However, qualitatively activation profiles shown in figure 3 in the original publication and those shown in figure 1 show good correspondence. A noticeable difference is that rectification in the original publication would prohibit neuron activations from exhibiting negative values. In our implementation negative values are possible and clearly exhibited by EBNs. Since we apply a rectified linear gain function, however, these negative values have a negligible effect on the behavior of the system as a whole.

The second simulation shows the relation between input strength and burst amplitude for LLBNs and EBNs. Inputs to the left SG were equal to $\mathrm{I} = 1$ (figure 2, blue),

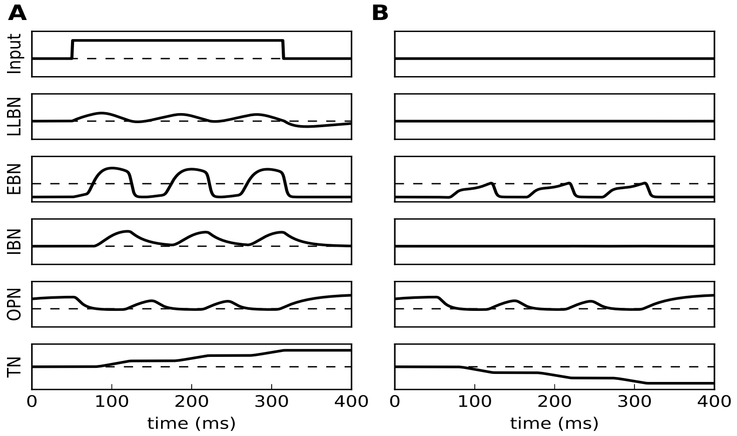

Figure 1: Activity profiles in the left (A) and right (B) SG. All activities are in response to constant input applied to the left LLBN. The dashed horizontal line marks zero. Excitatory burst neurons (EBNs) exhibit negative baseline activity.

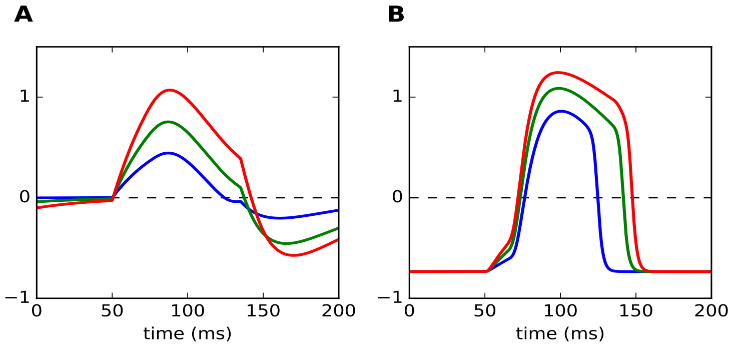

Figure 2: Activity profiles in LLBN (A) and EBN (B). Increased input strength results in larger LLBN and EBN burst size with inputs equal to 1 (blue), 1.75 (green), and 2.5 (red).

$\mathrm{I} = 1.75$ (figure 2, green), and $\mathrm{I} = 2.5$ (figure 2, red); each applied to the left LLBN for 85 ms. Qualitatively, our results accord well with those shown in figure 5 of the original publication. As before, our results differ only with respect to the possibility of negative activity values.

The third simulation produces saccades in response to different input strengths to the horizontal and vertical SG. Specifically, inputs to the right and upward LLBNs were ($\mathrm{I_r} = 0.67$, $\mathrm{I_u} = 0.08$; figure 3, blue), ($\mathrm{I_r} = 0.7$, $\mathrm{I_u} = 0.22$; figure 3, green), ($\mathrm{I_r} = 0.74$, $\mathrm{I_u} = 0.4$; figure 3, red), ($\mathrm{I_r} = 0.75$, $\mathrm{I_u} = 0.6$; figure 3, turqoiuse), and ($\mathrm{I_r} = 0.7$, $\mathrm{I_u} = 0.9$; figure 3, purple). Inputs to the horizontal and vertical SG were applied for 75 ms. Our results are again in good agreement with those shown in figure 6 of the original publication.

The fourth simulation generates a staircase of three saccades in response to continuous input. Here, inputs to the right and upward LLBNs were equal to $\mathrm{I_r} = 0.2$ and $\mathrm{I_u} = 0.33$ for 300 ms. In the original implementation, the input was presented for 250 ms. However, this lead to the production of two rather than three saccades in our implementation motivating the choice of a slightly longer stimulation period. Another difference is that our implementation produces slightly larger saccades. Specifically, after the first saccade horizontal and vertical eye positions are respectively at 4° and 6° in our simulation while they are at $\sim 3°$ and $\sim 5°$ in the original publication. Similarly, after the second saccade horizontal and vertical eye positions are respectively at 7° and 12° in our simulation while they are at $\sim 6°$ and $\sim 9°$ in the original publication.

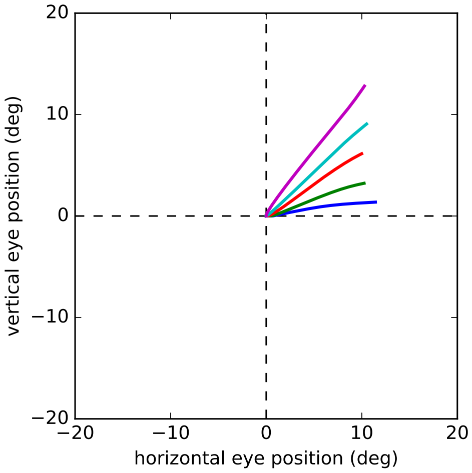

Figure 3: Oblique saccades. Saccades are fairly straight with a slight tendency to curve.

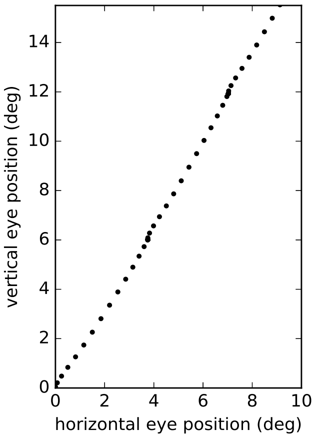

Figure 4: Saccadic staircase. Three saccades in a staircase continue in the same direction as the initial saccade. Eye position was sampled every $2\,\mathrm{ms}$.

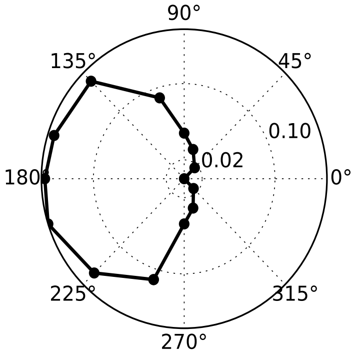

Figure 5: EBN tuning curve. Tuning curve of the left EBN exhibiting a cardioid-like shape.

In the fifth simulation the average activity of the left EBN is obtained for a series of saccades with different directions. Note that for reasons of comparability with the original publication, only positive activation values were averaged. Figure 5 shows a polar plot of average activity corresponding to each saccade reflecting the neuron's tuning curve. The inputs to the SG producing the desired saccades can be found in table 1. Each of these inputs was applied for 50 ms. The tuning curve we observe for the left EBN exhibits a cardioid-like shape as was the case in the original publication.

Table 1: Direction specific inputs to SG to produce EBN tuning curve.

	0	45	72	90	108	135	162	180	198	225	252	270	288	315
I_l	.00	.00	.00	.00	.20	.45	.63	.70	.63	.45	.20	.00	.00	.00
I_r	.70	.45	.20	.00	.00	.00	.00	.00	.00	.00	.00	.00	.20	.45
I_d	.00	.00	.00	.00	.00	.00	.00	.00	.20	.45	.63	.70	.63	.45
I_u	.00	.45	.63	.70	.63	.45	.20	.00	.00	.00	.00	.00	.00	.00

The sixth simulation reported in the original publication is designed to replicate results of Stanford *et al.* [6]. These authors stimulated the superior colliculus (SC) at various frequencies and measured the resulting saccade amplitude, duration, and velocity; showing that ampltidude saturates before velocity. Our implementation of the SG model was capable of replicating these results. However, reproducing simulation results reported by Gancarz & Grossberg [2] with our implementation was complicated by the fact that the stimulation protocol given by the authors lead to the production of two rather than a single saccade. Furthermore, the authors did not report their criteria for identifying saccade on- and offsets. Stanford *et al.* [6] used velocity criteria to determine onset ($v > 30$ deg/s) and offset ($v < 30$ deg/s) of a saccade. While this provided us with explicit criteria, velocity did not always drop below 30 deg/s after the first saccade before rising again with the second. To determine the offset of a saccade in those cases, we found the local minimum between the end of the first and the beginning of the second saccade. With these criteria in place, we stimulated the SC at a frequency varying between 1 and 2.4 at increments of 0.2. The connection weight from SC to LLBN was set equal to two and stimulation duration was 125 ms. Results of our simulation are shown in figure 6 and accord well with those reported in the original publication (their figure 9). Interestingly, saccade amplitude in response to a stimulation frequency of 1.2 in the original implementation is larger than expected given both our simulation and empirical results. It is possible that the offset criterion employed by Gancarz & Grossberg [2] failed to differentiate between two saccades following each other in very short succession leading to an overestimation of saccade duration.

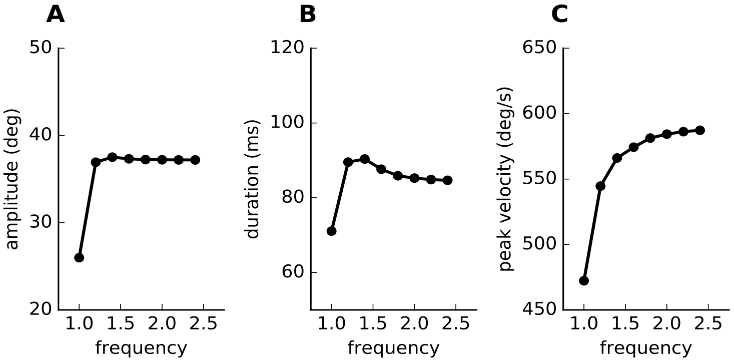

Figure 6: Effect of stimulation frequency on saccade (A) amplitude, (B) duration, and (C) velocity. Saccade amplitude (A) saturates before saccade velocity (C).

The seventh simulation shows that saccade velocity and duration can be traded while keeping amplitude constant. To produce a high-velocity saccade, the SC was stimulated with a frequency F = 3 for 62 ms. Similarly, to produce a low-velocity saccade, the SC was stimulated with a frequency of F = 1.3 for 112 ms. The stimulation duration was in both cases shorter than those reported by Gancarz & Grossberg [2] (82 ms and 117 ms) to reproduce the behavior of the SC shown in their figure 10 more faithfully. Figure 7 shows the results of our simulation. In accordance with the original simulation, saccade amplitude reflected by TN activity was identical after high- and low-frequency saccades. However, our implementation produced a second, small, saccade not observed for the original implementation.

The eighth simulation reported by Gancarz & Grossberg [2] shows how strong (I = 3) sustained (300 ms) input to the SG produces a single large saccade followed by a series of small saccades resembling smooth eye movements. Our results, shown in figure 8, reproduce these findings as they strongly resemble those shown in figure 11 of the original publication.

The final simulation showcases the evolution of activity exhibited by SG neurons when the OPN was electrically stimulated (J = 1.8) for 5 ms while a constant input (I = 0.7) was applied to the LLBN for 100 ms. External stimulation temporarily restored activation in the OPN and hence also inhibition of the EBN, leading to an interruption of the saccade. As is shown in figure 9, the saccade remained accurate despite this disruption. This is in agreement with results shown in figure 12 of the original publication.

Conclusion

The reproduced results generally show good qualitative correspondence with those reported by Gancarz and Grossberg [2] and accorded well quantitatively whenever such information was available. One notable difference is that EBNs in our implementation exhibited negative activity levels at baseline since we did not artificially bound them from below at zero. While this constitutes a significant change to the model, any effects this might have had were offset by passing inputs to each neuron through a rectified linear gain function rather than summing them linearly. furthermore, our implementation produced saccades of slighly larger amplitude than the original implementation. This was likely due to the different choice of gain function applied to the inputs of the OPN. Specifically, our gain function has a steeper slope which caused the OPN to shut-off faster in response to LLBN activty which, in turn, lead to prolonged release from inhibition of EBNs. These effects were minute but sufficient to cause differences in saccade amplitude of $\sim 1°$.

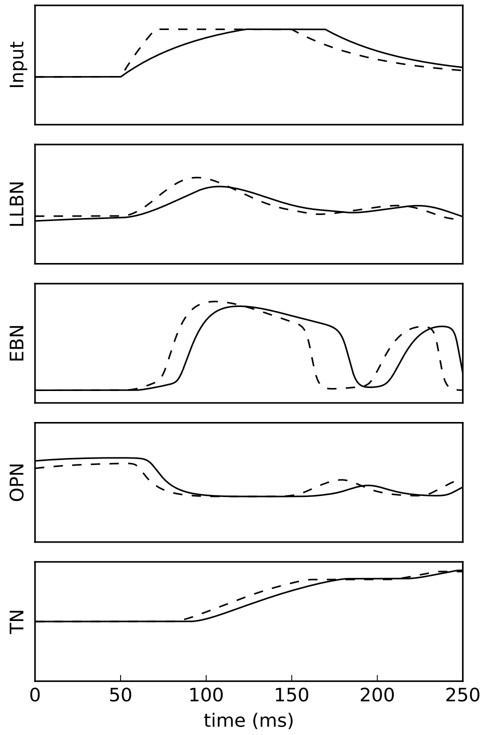

Figure 7: Trading saccade velocity for duration. Duration and velocity of a saccade can be traded while keeping amplitude constant given an appropriate shape of the input signal.

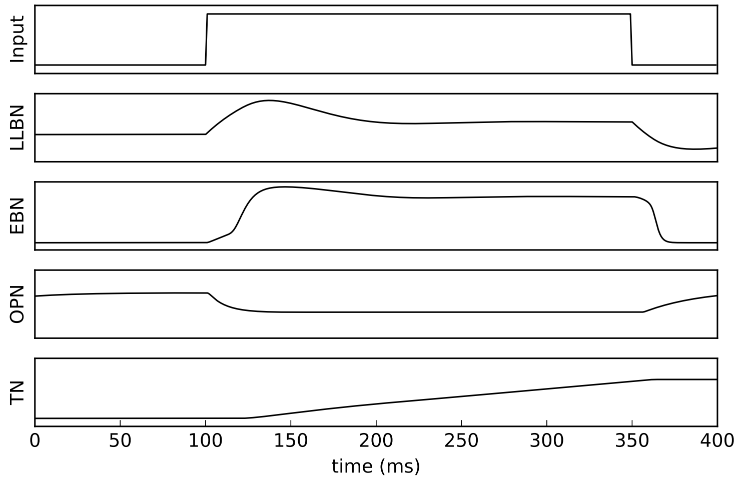

Figure 8: Smooth staircase eye movements. Activity profiles of SG neurons accompanying a single saccade followed by smooth eye movement as a result of strong sustained input.

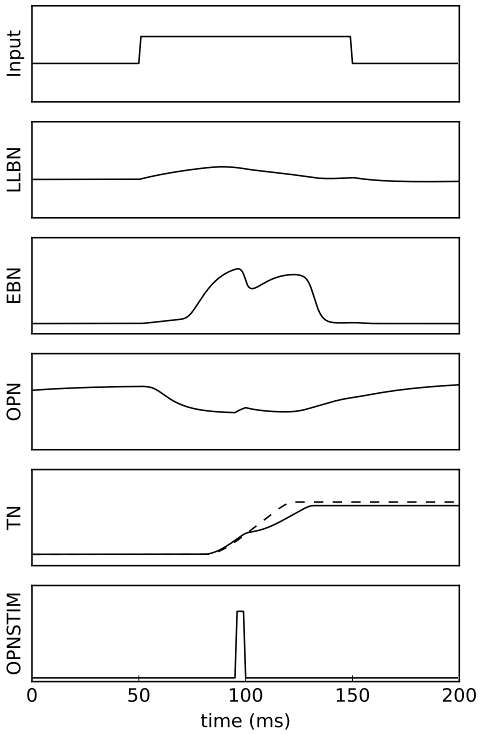

Figure 9: Interrupted saccade resulting from OPN stimulation. OPN stimulation interrupts the saccade, which remains accurate nonetheless. The dashed line shows TN activity for an uninterrupted saccade.

In conclusion, our reproduction confirms the results of the original publication and shows that an implementation in the NEST framework is feasible. This allows for the straightforward integration of the saccade generator into models of the visuomotor system forming the interface between sensory processing and motor control.

Acknowledgments

All network simulations carried out with NEST (http://www.nest-simulator.org).

References

[1] Jochen M Eppler. "PyNEST: A convenient interface to the NEST simulator". In: *Frontiers in Neuroinformatics* 2 (2008). ISSN: 16625196. DOI: 10.3389/neuro.11.012.2008. URL: http://journal.frontiersin.org/article/10.3389/neuro.11.012.2008/abstract.

[2] Gregory Gancarz and Stephen Grossberg. "A neural model of the saccade generator in the reticular formation". In: *Neural Networks* 11.7 (1998), pp. 1159–1174. ISSN: 08936080. DOI: 10.1016/S0893-6080(98)00096-3.

[3] Marc-Oliver Gewaltig and Markus Diesmann. "NEST (NEural Simulation Tool)". In: *Scholarpedia* 2.4 (2007), p. 1430. ISSN: 1941-6016. DOI: 10.4249/scholarpedia.1430. URL: http://www.scholarpedia.org/article/NEST%7B%5C_%7D%7B%%7D28NEural%7B%5C_%7DSimulation%7B%5C_%7DTool%7B%%7D29.

[4] Stephen Grossberg, Krishna Srihasam, and Daniel Bullock. "Neural dynamics of saccadic and smooth pursuit eye movement coordination during visual tracking of unpredictably moving targets". In: *Neural Networks* 27 (2012), pp. 1–20. ISSN: 08936080. DOI: 10.1016/j.neunet.2011.10.011.

[5] Jan Hahne et al. "Integration of continuous-time dynamics in a spiking neural network simulator". In: (Oct. 2016). arXiv: 1610.09990. URL: http://arxiv.org/abs/1610.09990.

[6] T R Stanford, E G Freedman, and D L Sparks. "Site and parameters of microstimulation: evidence for independent effects on the properties of saccades evoked from the primate superior colliculus." In: *Journal of neurophysiology* 76.5 (Nov. 1996), pp. 3360–81. ISSN: 0022-3077. URL: http://www.ncbi.nlm.nih.gov/pubmed/8930279.