Spike Train Generation

Jaeho Cho

MA-391 - Medical Imaging

The Cooper Union for the Advancement of Science and Art

New York City, NY

jaeho.cho@cooper.edu

Abstract—**This project presents an implementation and analysis of spike-train generation using Poisson processes with and without refractoriness, following concepts from *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems* (Dayan & Abbott, 2001). The characteristics of stochastic spike sequences are examined through statistical measures including the Fano factor, interspike interval (ISI) distributions, coefficient of variation (CV), and spike-train autocorrelation functions. The study explores both homogeneous and inhomogeneous Poisson processes and incorporates refractory effects using absolute and relative refractory periods.**

Index Terms—**Computational neuroscience, Stochastic processes**

I. Introduction

Neurons represent and transmit information by generating characteristic electrical pulses called action potentials or, more simply, spikes that can travel down nerve fibers. Ignoring the brief duration of an action potential (about 1 ms), an action potential sequence can be characterized simply by a sequence of spike times.

When generating a spike sequence, a stochastic process is often used to model the timing of spikes. In general, the probability of an event occurring at any given time could depend on the entire history of preceding events [1]. However, if there is no dependence on preceding events, so that the events are independent, the process is called a Poisson process.

The Poisson process provides a simple but powerful approximation of stochastic neuronal firing, however, certain features of neuronal firing violate the independence assumption of Poisson processes. One such feature is the refractory period, which is the time interval during which a neuron is unable to fire another action potential.

For a few milliseconds just after an action potential has been fired, it may be virtually impossible to initiate another spike, this is called the absolute refractory period. For a longer interval known as the relative refractory period, it is more difficult to evoke an action potential.

II. Methodology

This project implemented the generation of spike trains in a Python notebook with a python3.12 interpreter and utilized the numpy library to support computations. Two methods of spike generation from the textbook were explored: a simple generator and a fast generator.

A. Simple Poisson Spike Generator

The simple spike generator follows the initial approach outlined in the textbook. Given a firing rate, r or $r_{est}(t)$, the probability of firing a spike in a small time interval Δt is computed as $r_{est}(t)\Delta t$. The algorithm iterates through discrete time steps of size Δt, generating a random number x_{rand} drawn from a uniform distribution between 0 and 1 at each step. A spike is fired if $r_{est}(t)\Delta t > x_{rand}$, otherwise, no spike is generated.

The generator naturally supports both homogeneous and inhomogeneous Poisson processes and also incorporates refractory effects. An absolute refractory period, t_{abs}, prevents spikes from occurring within a fixed interval after a spike, and a relative refractory period, modeled as an exponential distribution with mean t_{rel}, further modulates the likelihood of a spike. The spike sequence is stored as a binary vector, where a value of 1 denotes a spike occurrence in the corresponding time bin.

B. Fast Interspike Interval Spike Generator

A faster spike generation method, leveraging interspike interval (ISI) sampling, is used to improve computational efficiency. For homogeneous Poisson processes, spike times t_i are generated iteratively using exponentially distributed ISIs. Given a firing rate r, the ISIs follow the probability density function:

$$P(\text{ISI} = \tau) = re^{-r\tau} \tag{1}$$

which can be sampled by inverting the cumulative distribution function:

$$\tau = -\ln(x_{rand})/r \tag{2}$$

The spike times are then computed as the cumulative sum of ISIs.

Due to the generation of all ISIs in advance, the method is faster than the simple generator but results in an unknown duration of the spike sequence. To address this, an excess number of ISIs are generated, and the spike sequence is truncated to the desired duration.

For inhomogeneous Poisson processes, rejection sampling (spike thinning) is applied. The method requires an upper bound on the firing rate, r_{max} for the initial generation of a spike sequence using the previously mentioned method. Each generated spike t_i is discarded with probability $(1 - r_{est}(t_i)/r_{max})$ ensuring that the final spike sequence adheres to the time-varying firing rate.

To incorporate refractory effects, an additional thinning step is introduced. For each spike, a total refractory period $t_{\text{refract}} = t_{\text{abs}} + t_{\text{rel}}$ is defined, where t_{abs} is a constant absolute refractory period and t_{rel} is the relative refractory period that is drawn from an exponential distribution. The spike sequence is then thinned by removing spikes that occur within the total refractory period of the spike in question.

Both the simple and fast generators produce spike trains with theoretically identical statistical properties, but differ in efficiency and precision. The simple generator operates on fixed time steps, limiting temporal resolution to Δt. In contrast, the fast generator allows continuous spike times, achieving precision up to 17 decimal places [2].

C. Analysis

For the statistical analysis of spike sequences, 5000 trials of 1 s duration were generated using the fast generator for all four variations of the Poisson process: homogeneous, inhomogeneous, homogeneous with refractoriness, and inhomogeneous with refractoriness. The raster plots over 10 of these trials are presented in Fig. 1.

For the purposes of comparison, the homogeneous firing rate was set to the max of the inhomogeneous firing rate 60 Hz, and the refractory periods were set to 1 ms and 10 ms for the absolute refractory period and relative refractory period mean, respectively [3]. The inhomogeneous firing rate was modeled as a step function with equally spaced intervals as illustrated in Fig. 2.

The following statistical measures were computed: Fano factor, interspike interval (ISI) distributions, coefficient of variation (CV), spike-train autocorrelation functions.

To compute the spike-train autocorrelation functions, a method similar to circular convolution is employed, though it is not mentioned in the textbook. This approach involves constructing a periodic spike sequence and determining all possible intervals between spikes, including both wrap-around and self-intervals. The autocorrelation function is then derived by counting the number of spike intervals that fall within time
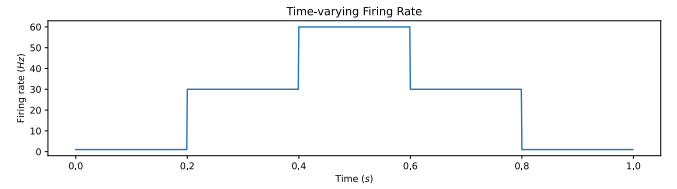

Fig. 2: The time-varying firing rate $r_{\text{est}}(t)$ used for the inhomogeneous process, with firing rates of $1, 30, 60, 30, 1$ Hz at intervals of 0.2 s.

bins defined by $(m - 1/2)\Delta t$ and $(m + 1/2)\Delta t$, where m is an integer (positive or negative) representing the bin index of the autocorrelation function.

III. Results

The computed Fano factors of the processes are presented in Table I along with the variance σ_n^2 and trial average of the spike counts. The Fano factor of the homogeneous poisson process is expectedly 1 and so is the Fano factor of the inhomogeneous poisson process. The processes including refractory effects on the other hand are closer to 0.5.

TABLE I: Fano Factor

Process	σ_n^2	$\langle n \rangle$	Fano Factor
Homogeneous	59.77	59.82	1.00
Homogeneous w/ Refractory Effects	17.20	36.26	0.47
Inhomogeneous	24.47	24.45	1.00
Inhomogeneous w/ Refractory Effects	9.66	16.73	0.57

The ISI probability distributions shown in Fig. 3 match the expected exponential distribution for the Poisson processes. The processes without refractory effects closely resemble an exponential distribution, while refractory effects cause the distributions to have a significant drop in the probability of short ISIs.
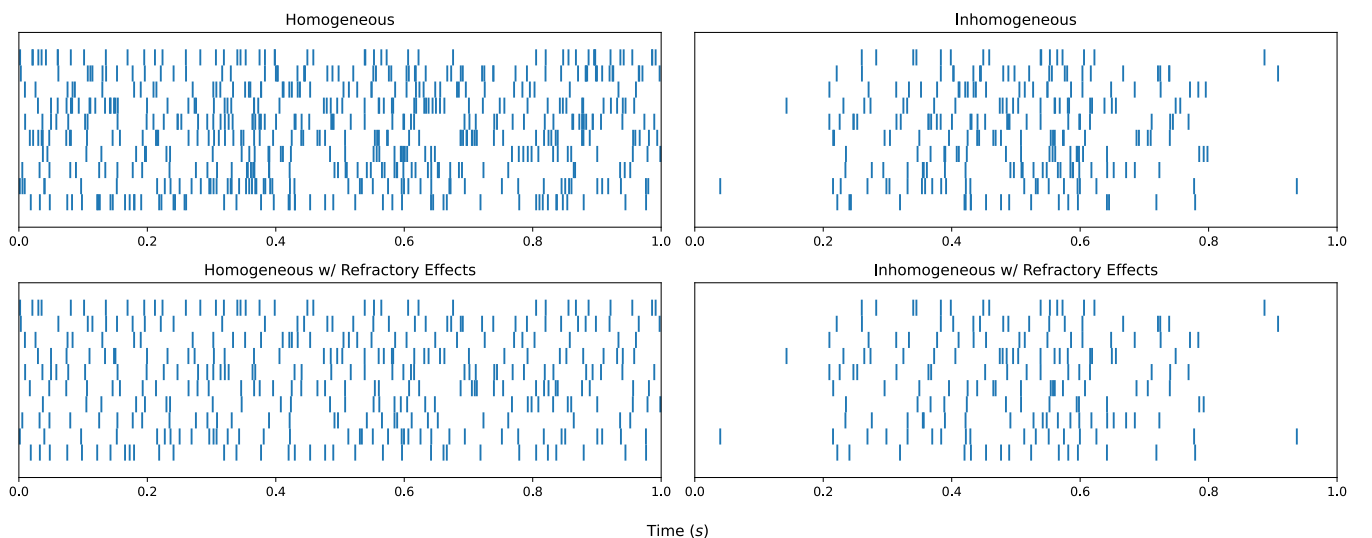

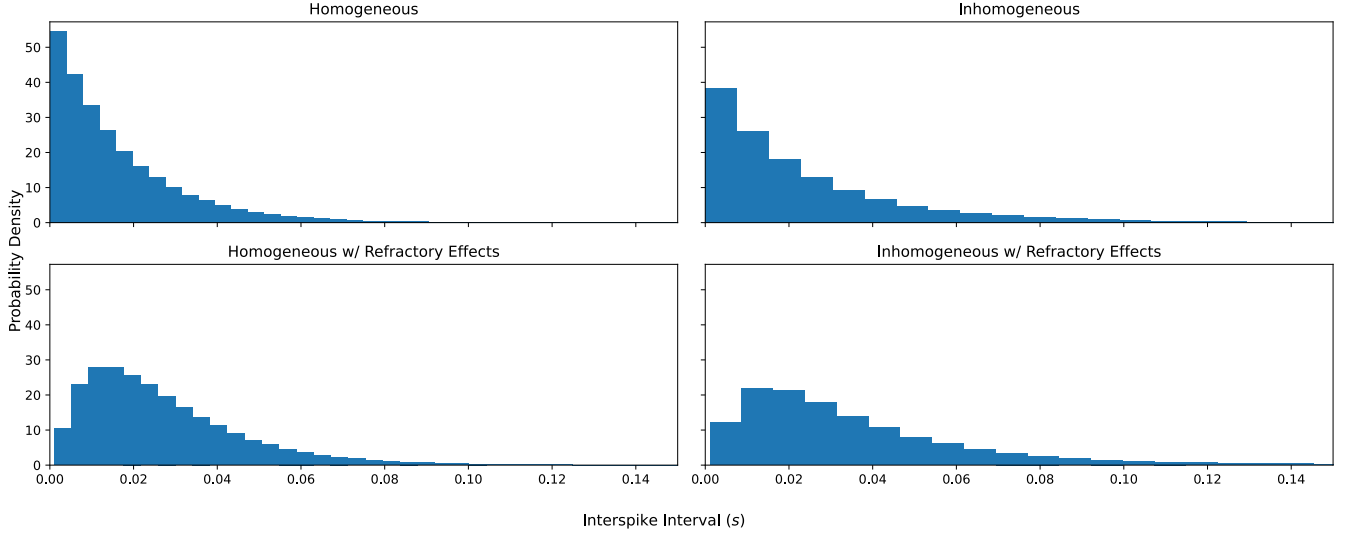Fig. 1: Raster plots for 10 trials.

Fig. 3: Interspike interval distributions. The histograms were generated with 50 bins and scaled to share the x-axis.

The coefficients of variation are presented in Table II along with the standard deviation σ_τ and trial average of the interspike intervals. As expected, the coefficient of variation is 1 for the homogeneous poisson process but is noticeably higher for the inhomogeneous poisson process. The processes with refractory effects have noticeably lower coefficients of variation.

TABLE II: COEFFICIENT OF VARIATION C_V

Process	σ_τ	$\langle\tau\rangle$	C_V
Homogeneous	0.016	0.016	1.00
Homogeneous w/ Refractory Effects	0.019	0.027	0.70
Inhomogeneous	0.029	0.025	1.17
Inhomogeneous w/ Refractory Effects	0.032	0.037	0.87

Fig. 4 shows as expected, the spike-train autocorrelation function of the homogeneous poisson process is a δ function at $\tau = 0$ with a value equal to the average firing rate. The autocorrelation functions of the inhomogeneous poisson process and the processes with refractory effects are also shown in Fig. 5. The autocorrelation function for the homogeneous process is uniformly distributed as expected, while the inhomogeneous process has a peak at $\tau = 0$ and decreases linearly as the τ increases. The processes with refractory effects have a similar shape to the inhomogeneous process but with a drop near $\tau = 0$.

IV. DISCUSSION

Though the ISI spike generator is faster at generating homogeneous spike sequences, it is still limited by the iterative processes of spike thinning and could be optimized further.

The effects of inhomogeneous firing rate and refractory effects are made clear in Fig. 6 where each process shares an initial homogeneous sequence. The method of spike thinning with the fast generator allows for the direct comparison

between homogeneous and inhomogeneous processes, such that the spikes that are removed are clearly addressable.

The refractory periods are also evident in the raster plots of Fig. 6, where the refractory periods are highlighted in red. Comparing the raster plots of the processes without refractory periods to their corresponding plots with, it is evident that the refractory periods cause the spikes to be more equally distributed, which is reflected in the Fano factors.

A surprising effect of the relatively longer relative refractory period mean compared to the absolute refractory period is that the refractory period is dominated by the relative refractory period. This is evident in Fig. 6 where some refractory periods are unnoticeable but others effectively remove multiple spikes.

I suspect that the Fano factor for the inhomogeneous process is 1 because the time-varying spike rate used for the inhomogeneous process is consistent through trials, thus the variance and mean of the spike count would remain equal. Since the Fano factor is the ratio of the variance to the mean number of spikes, a Fano factor of 0.5 indicates that the variance is lower than expected or the mean is higher than expected. Since refractory effects should only reduce the mean number of spikes, the former is more likely; I suspect that the refractory period is causing the spikes to be more evenly distributed, reducing the variance.
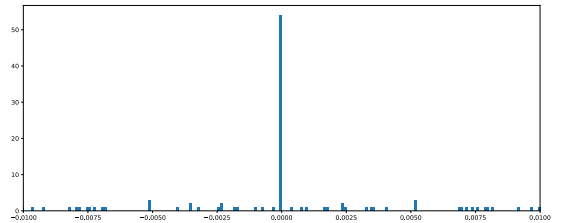

Fig. 4: Autocorrelation Function of a homogeneous poisson process that generated 54 spikes, with $\Delta t = 0.0001 \approx 0$ such that the number of spikes in the $m = 0$ bin is also 54.
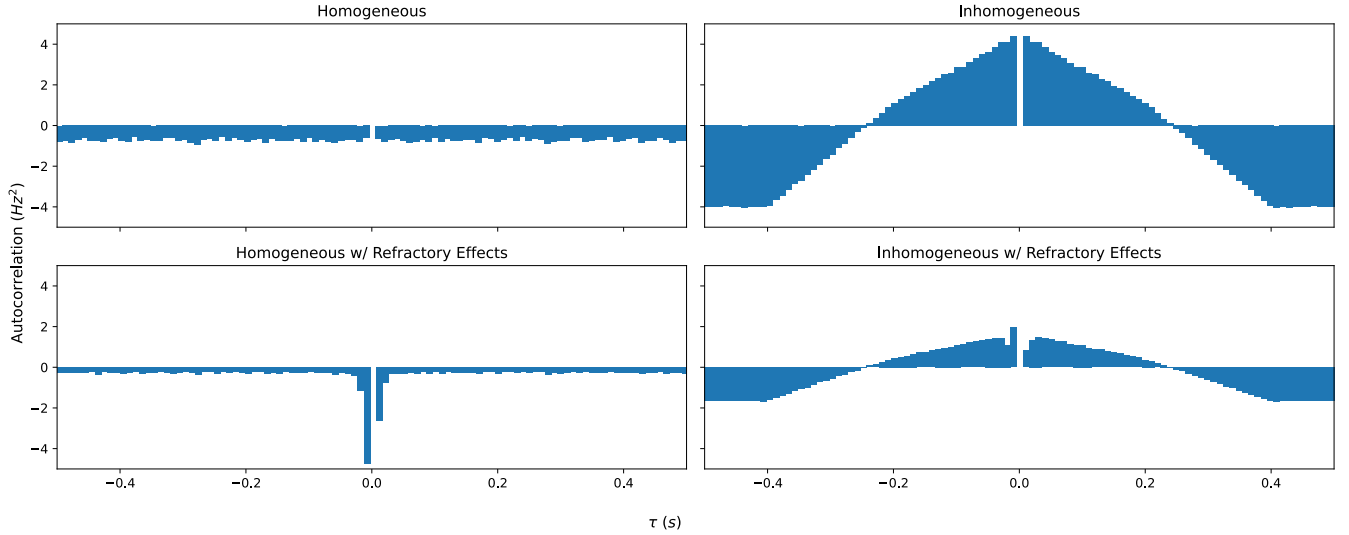
Fig. 5: Spike-Train Autocorrelation Functions with $m = 0$ bin removed.

In the ISI distributions, refractory effects prevent spikes from occurring too close together, resulting in the significant drop in the probabilities of short ISIs for both homogeneous and inhomogeneous spike processes.

The increase in the coefficient of variation for the inhomogeneous process is understandable as the time-varying firing rate allows for more variation in the ISIs. Also, the coefficient of variation for the processes including refractory effects are below 1 indicating the standard deviation is lower or the mean is higher than without refractory effects. Since refractory periods remove spikes with a short interspike interval, the processes should increase the mean interspike interval.

Due to the half-open bins in `np.histogram`, the autocorrelation function is not perfectly symmetric [4]. It may be possible to improve the implementation of the spike-train autocorrelation function by using convolution or correlation functions in the future. For the trial averaged spike-train auto-

correlation function, another method considered was creating a concatenated spike train of all trials, and computing the autocorrelation function by considering all pairs of spikes within the trial duration as a sliding function, but was not implemented for this project as it would require a large amount of memory for a large number of trials and may not be what the function is meant to represent.

V. Appendix

During the writing of the spike-train autocorrelation function, a mistake in the implementation was identified where the function was computing the autocorrelation of the interspike intervals. The results of the computation are presented in Fig. 7. Though I am unsure how to interpret the results of this computation, it may have meaningful insights that warrant further exploration.
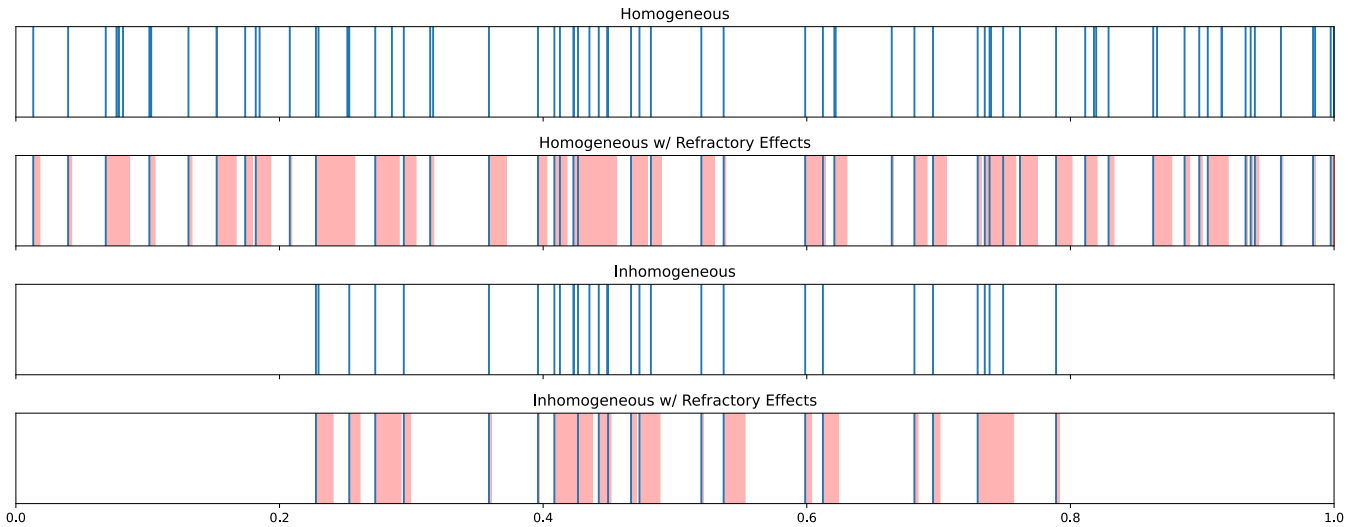

Fig. 6: Raster Plots of the Homogeneous and Inhomogeneous Processes with and without Refractory Effects. The processes with refractory effects have their refractory periods highlighted in red.
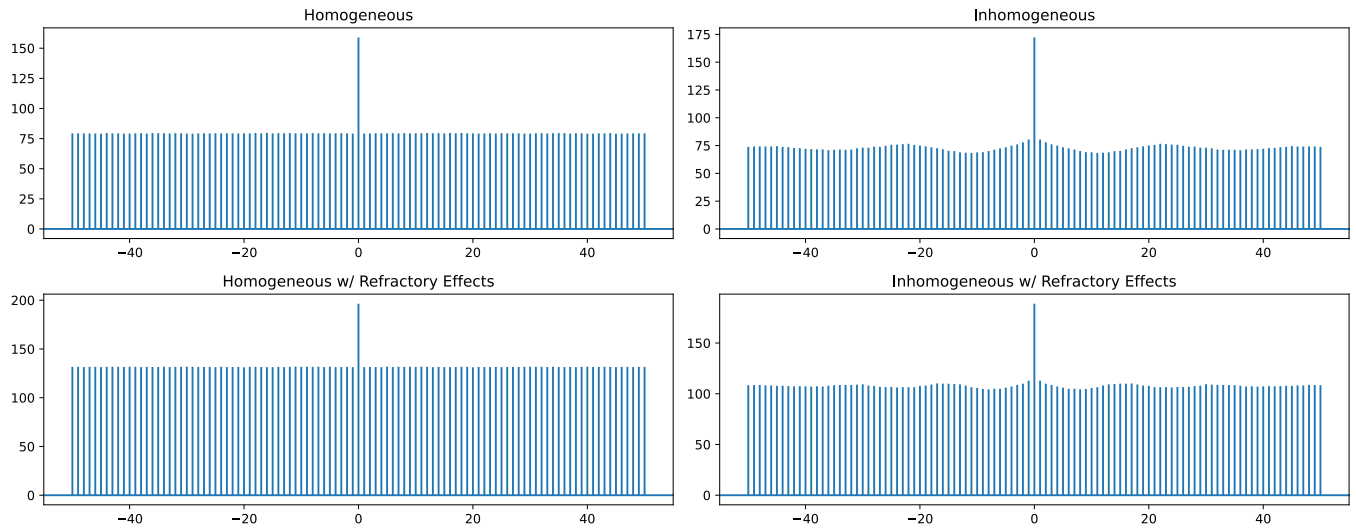
Fig. 7: Autocorrelation functions of the interspike intervals

REFERENCES

[1] P. Dayan and L. F. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. in Computational Neuroscience. Cambridge, Mass.: MIT Press, 2001.

[2] Python, "Floating-Point Arithmetic: Issues and Limitations." Accessed: Feb. 16, 2025. [Online]. Available: https://docs.python.org/3/tutorial/floatingpoint.html

[3] F. Gabbiani and S. J. Cox, *Mathematics for Neuroscientists*, 1st ed. Amsterdam: Elsevier/Academic Press, 2010.

[4] "Numpy.Histogram — NumPy v2.2 Manual." Accessed: Feb. 19, 2025. [Online]. Available: https://numpy.org/doc/stable/reference/generated/numpy.histogram.html