PYTHON EXERCISE I FOR SC3011TN: ACADEMIC YEAR 2020/2021

Python Exercise I: Simulating the Motion of a Black Hole

Carlas Smith, Michel Verhaegen, Dylan Kalisvaart & Jacques Noom

Version 1.1

February 10, 2021

Contents

Chapter 1

Introduction

1.1 Brownian Motion & Black Hole Dynamics

Brownian motion was first observed experimentally by Robert Brown in 1827 [2]. He used a simple microscope to study the movement of particles from pollen in water immersion. He observed that pollen particles of the size of $5\mu m$ were not at rest.

In modern physics, Brownian motion plays an increasingly important role. Brownian motion is for instance observed in nonequilibrium systems at small scale related to the transport of molecules and cells in biological systems, as well as at macroscale related to stellar dynamics of a massive pointlike object, such as a black hole near the center of a dense stellar system [3].

Brownian motion is also an interesting phenomenon to introduce crucial concepts of *stochastic processes* and the simulation of stochastic differential equations. Though the latter requires advanced mathematical foreground knowledge, we will take a more pragmatic view in order to simulate such systems.

As an example of a dynamical system governed by Brownian motion, you will analyze the dynamics of a black hole in this assignment. Specifically, the 1-dimensional dynamics (in the x-direction) of a massive black hole at the center of a dense stellar system (which is distributed according to a Plummer model) can be described by a driven harmonic oscillator, where the driving force is represented by a random variable [3]. The continuous-time dynamics are given by the the following Langevin equation [4, 5]:

$$m\frac{d^2x(t)}{dt^2} + \gamma\frac{dx(t)}{dt} + cx(t) = \sqrt{R}\,w(t) \tag{1.1}$$

The variables in this equation are defined in Table 1.1. In order to simulate this process, it is important to know how to generate the white noise sequence $w(t)$ in a sampled data setting. For that purpose we discuss physically realizable white noise sequences first. This is done in Section 1.2.

Table 1.1: The quantities in the Langevin equation (1.1).

Quantity	Meaning
m	mass of the black hole
γ	friction coefficient
c	spring constant
R	noise factor
$x(t)$	center displacement of the black hole in the x-direction at time instant t
$w(t)$	normalized white noise: $\forall t, t' : E[w(t)] = 0 \;\; E[w(t)w(t')] = \delta(t - t')$.
$\delta(t)$	the dirac delta function

1.2 Bandlimited White Noise

The quantity $w(t)$ in Table 1.1 is continuous white noise. If we consider the spectrum as the Fourier transform of the autocorrelation function, the variance of white noise is *infinity*. As such, white noise is physically not realizable. In practice the notion of *bandlimited white noise* is used. Such signal has the following spectrum:

$$P_w(\omega) = \begin{cases} \sigma & \text{for } |\omega| < \omega_0 \\ 0 & \text{otherwise} \end{cases} \tag{1.2}$$

The autocorrelation function of a bandlimited white noise signal is:

$$r_w(\tau) = \int_{-\omega_0}^{\omega_0} \sigma e^{-j\omega\tau} d\omega = 2\sigma \frac{\sin(\omega_0 \tau)}{\tau} \tag{1.3}$$

We plot this autocorrelation function for three values of ω_0 in Figure 1.1.

Since the limit $\lim_{t\to 0} \frac{\sin(\omega_0 t)}{t} = \omega_0$, it can be observed that the value $r_w(0)$ increases with ω_0. This means that the variance increases linearly with ω_0, i.e. the bandwidth of the signal. In the limit as the bandwidth goes to infinity we have,

$$\lim_{\omega_0 \to \infty} r_w(\tau) = 2\pi\sigma\delta(\tau)$$

For an example of physically generating bandlimited white noise, we refer to Example 1.1.

Example 1.1 (Noise generated by an RC circuit. [1])
Consider an RC circuit with transfer function given as:

$$G(j\omega) = \frac{a}{\sqrt{\pi}(j\omega + a)}$$

The spectral density of white noise filtered by this transfer function is[1]:

$$P(\omega) = \frac{a^2}{\pi(\omega^2 + a^2)}$$

[1]See (3.91) on page 101 of [7] for the definition (in the discrete case).

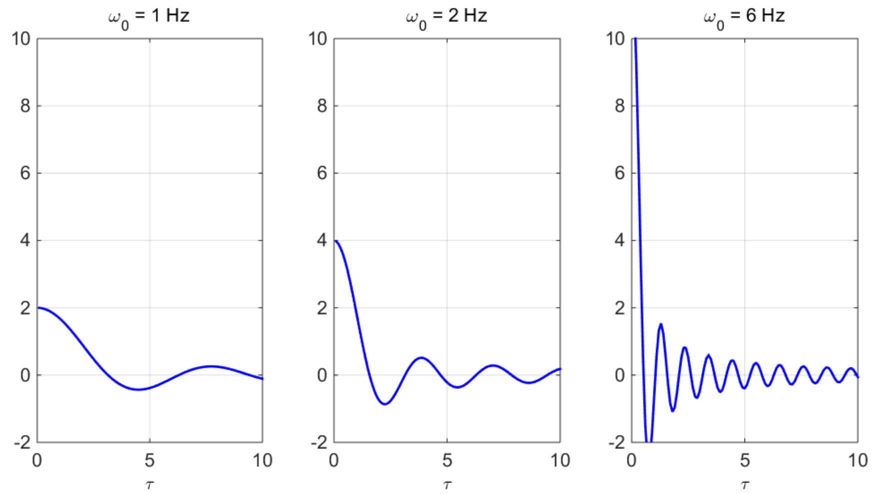

Figure 1.1: The autocorrelation function for three bandlimited white noise signals for ω_0 resp. equal to $1, 2, 6$ Hz.

For $|\omega| < a$ the spectral density is essentially constant. The autocorrelation function that result as the inverse Fourier transform of this spectral density equals:

$$r(\tau) = ae^{-a|\tau|}$$

Therefore we have the following limits:

$$\lim_{a\to\infty} P(\omega) = \frac{1}{\pi} \qquad \lim_{a\to\infty} r(\tau) = \delta(\tau)$$

That means that for increasing value of a, the filtered white noise signal by the RC filter will resemble more and more the spectral properties of white noise. It will also have an increasing variance.

1.3 Simulating the Langevin Equation

As outlined in Section 1.2, the variance of a physically realizable white noise signal increases with the bandwidth ω_0. In order to analyse the dynamics of a black hole (1.1) in a discrete time setting, we consider the use of a sampling period Δt sec.

When considering the Discrete Fourier Transform [7], the maximal frequency value that is obtained for a sampled data sequence with sampling period Δt is proportional to $\frac{1}{\Delta t}$. Based on this fact and the increase of the variance of bandlimited (physically realizable) white noise with the bandwidth, a discrete counterpart for the white noise signal in the Langevin equation (1.1) is chosen as follows [6]. The discrete sequence of random numbers $\{\breve{w}(k)\}$ that mimics the properties of $w(t)$ in (1.1) should have mean zero and variance that is equal to $\frac{1}{\Delta t}$. Such a sequence is called discrete white noise [7].

This concept can be used to replace white noise in discretizing different stochastic differential equations [6] and is illustrated in the following RC example.

Example 1.2 (Simulating the noise of the RC circuit of Example 1.1)

When consider the signal $y(t)$ to be generated by filtering the white noise signal $w(t)$ defined in Table 1.1 by the continuous-time filter with transfer function:

$$G(s) = \frac{a}{\sqrt{\pi}(s+a)}$$

it should satisfy the stochastic differential equation:

$$\dot{y}(t) + ay(t) = \frac{a}{\sqrt{\pi}}w(t) \quad y(0) = y_0 \tag{1.4}$$

In order to avoid the use of advanced mathematical concepts as Ito Integrals, martingales, etc., we make use of the simplified (engineering) approach outlined in [6] to transform the stochastic differential into a difference equation. For that purpose a discrete white noise sequence $\tilde{w}(k)$ is generated with the following properties (where $\breve{w}(k)$ denotes the discretized white noise signal):

$$\begin{aligned} & \tilde{w}(k) = \breve{w}(k)\sqrt{\Delta t} \\ \forall k, \ell : E[\tilde{w}(k)] = 0 \quad & E[\tilde{w}(k)\tilde{w}(\ell)] = \begin{cases} 1 & k = \ell \\ 0 & k \neq \ell \end{cases} \end{aligned} \tag{1.5}$$

When the discrete white noise is selected in this way, the stochastic differential equation could be transformed into a (first order) difference equation by applying Eulers backward approximation to the derivative[2]. If the continuous time variable $y(t)$ is approximated by the discrete-time sequence $y(k\Delta t)$ (or in short simply denoted by $y(k)$), then difference equation that results by discretizing the stochastic differential equation (1.4) becomes:

$$\begin{aligned} \frac{y(k) - y(k-1)}{\Delta t} + ay(k) &= \frac{a}{\sqrt{\pi}} \frac{\tilde{w}(k)}{\sqrt{\Delta t}} \\ y(k) - \frac{1}{1 + a\Delta t} y(k-1) &= \frac{a}{\sqrt{\pi}} \frac{\sqrt{\Delta t}}{1 + a\Delta t} \tilde{w}(k) \end{aligned} \tag{1.6}$$

The approximation $y(k)$ is simulated in this way for three values of a, namely $1, 2$ and 5 for a discrete white noise sequence of 1000 samples. From this sequence, the autocorrelation function is approximated by the Python command `numpy.correlate`. The discrete differential equation (1.6) defines a discrete time system with transfer function $G(z)$ given as:

$$G(z) = \frac{a\sqrt{\Delta t}z}{\sqrt{\pi}(1 + a\Delta t)z - \sqrt{\pi}}$$

Here z is the z-transform variable equal to $z = e^{j\omega\Delta t}$ [7]

---

[2]The backward Euler method gives an approximation of a derivative via: $\dot{y}(t) \approx \frac{y(k)-y(k-1)}{\Delta t}$.

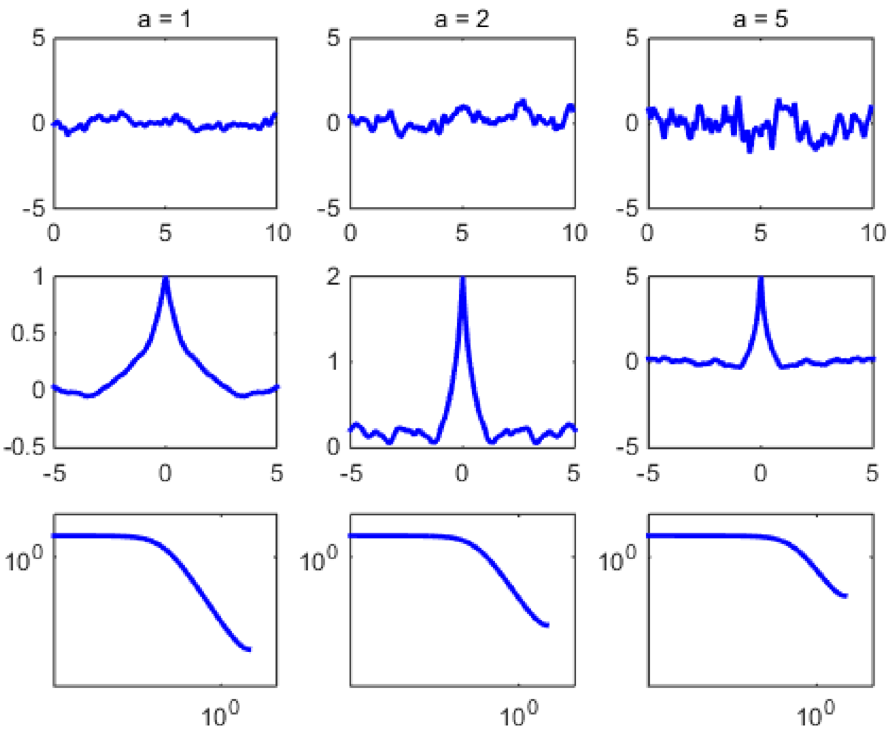

Figure 1.2: Simulating the discrete differential equation (1.6) for three different values of a equal to $1, 2$ and 5 resp. In the first row of figures one particular solution is plotted. The second row contains the approximation of the autocorrelation function by the Python command `numpy.correlate` and the third row contains the discrete power spectra.

Chapter 2

The Python Exercise

2.1 The Langevin Equation

In this exercise, we will analyze and simulate the one-dimensional dynamics of a black hole, to determine if the process is wide sense stationary (WSS). To be able to simulate the dynamics of (1.1), model parameters are given in Table 2.1.

Table 2.1: Model parameters of Langevin Equation (1.1).

Quantity	Meaning	Value
Δt	sampling period	10^{-3}
a	Plummer radius	$3\pi/16$
b_{max}	maximum impact parameter	$\approx a$
G	"normalized" gravitational constant	1
M	"normalized" total cluster mass	1
m_*	mass of an individual interacting star	M/N_*
N_*	number of interacting stars	first 5 digits of student number 1
V_0	relative velocity with interacting stars	$\approx \sqrt{GM/(2a)}$
Λ	Coulomb logarithm factor	$b_{\text{max}} V_0^2/(Gm)$
c	spring constant	GMm/a^3
m	mass of the black hole	0.01
R	noise factor	$4GMm_*\gamma/(9a)$
γ	friction coefficient	$\frac{128\sqrt{2}}{7\pi}\left(\frac{G}{Ma^3}\right)^{1/2} m^2 \log \Lambda$

2.2 Exercise

For simulating (1.1), the following exercises need to be solved. For your convenience, a Jupyter notebook is provided on Brightspace to help you complete the assignment and report about your findings. We strongly encourage using the notebook, as hints and code snippets are already provided for you.

1 point

1. If the random process $x(k)$ in (1.1) is wide-sense stationary (WSS), what can you say about its mean and variance? And what if the process is not WSS?

2. Let $x(k)$ be the discrete-time representation of the position of the black hole, with $t = k\Delta t$. Derive the discrete-time dynamics of $x(k)$. That is, find the parameters $\beta_1, \beta_2, \beta_3$ in the following second order difference equation:

$$x(k) + \beta_1 \cdot x(k-1) + \beta_2 \cdot x(k-2) = \beta_3 \cdot \tilde{w}(k) \tag{2.1}$$

Use Euler's backward approximation to approximate the derivative operator $\frac{d(.)}{dt}$ and second order derivative operator $\frac{d^2(.)}{d^2t}$ in (1.1). Subsequently replace the white noise signal $w(t)$ by a discrete white noise sequence as outlined in Section 1.3. Your answer should consist of two parts:

3 points (1) Analytical expressions for the parameters $\beta_1, \beta_2, \beta_3$

1 point (2) Their numerical values, by making use of the data given in Table 2.1.

3. Analyze the stochastic discrete-time dynamics in the z-domain. Specifically, answer the following questions:
(1) Using the z-transform, determine the transfer function $H(z)$ from $\tilde{w}$ to x of the discrete-time dynamical system described by difference equation (2.1). 1 point
(2) Determine the poles of $H(z)$. Is the system (BIBO) stable? Use Definition 2.5 or Lemma 2.6 on page 24 of [8]. 1 point

4. Simulate a single realization of the trajectory of the black hole. That is, simulate the difference equation (2.1) for $k = 3, \cdots, N$ with $N = 5000$ using the initial conditions $x(1) = x(2) = 0$. Hereby you should generate discrete white noise samples using the `numpy` command
`numpy.random.normal(size=N)`.
Your answer should consist of two parts:
1 point (1) Python code to generate N samples of $x(k)$
(2) Plot a realization of one sequence $x(k)$. Explain the results. Do the results agree with your analysis in Question 3? 1 point

5. In Question 4, you generated one realization of the trajectory of the black hole. To formalize this, let us denote the realization generated in Question 4 as $x(k, \lambda)$ for $\lambda = 1$ and the corresponding white noise sequence as $\tilde{w}(k, 1)$.

Now, simulate multiple realizations of the trajectory of the black hole. Specifically, generate L realizations $x(k, \lambda)$ for $\lambda = 1, \cdots, L$, with each realization generated for a different realization of the discrete time white noise sequence $\tilde{w}(k, \lambda)$.
Your answer should consist of two parts:
1 point (1) Python script used to generate L realizations $x(k, \lambda)$ sequences
(2) Plot of all L realizations for $L = 50$. Explain the results. Do the results agree with your analysis in Question 3? 1 point

6. Now, simulate even more realizations of the trajectory of the black hole.

   Increase L, as defined in Question 5, to respectively $L = 100,\ 500,\ 2500$. For each value of L, obtain all realizations at time steps $k = 10^3$, $k = 10^4$ and $k = 10^5$. That is, find $\{x(10^3, \lambda)\}_{\lambda=1}^{L},\ \{x(10^4, \lambda)\}_{\lambda=1}^{L},\ \{x(10^5, \lambda)\}_{\lambda=1}^{L}$.

   Using this data, generate histograms with `python` command `matplotlib.pyplot.hist()`, for each combination of time $k = 10^3, 10^4, 10^5$ and number of realizations $L = 100,\ 500,\ 2500$. Your answer should thus consist of 9 histograms. The number of bins in each histogram should be equal to $\sqrt{L}$.
   Provide one figure containing the 9 histograms. Make sure the figure has no overlapping text! Comment on your results. $\boxed{\text{2 points}}$

7. Fit a Gaussian function, given by $f(\alpha) = \kappa e^{-\frac{\alpha^2}{2\sigma^2}}$, to the 9 histograms Question 6. For that purpose, denote the center of bin i of the histogram by α_i and the corresponding height of the histogram as $h(\alpha_i)$. Then solve the following least squares optimization problem using the `scipy` function `scipy.optimize.least_squares`

$$\hat{\kappa}, \hat{\sigma} = \arg\min_{\kappa,\sigma} \sum_{i=1}^{P} \left(h(\alpha_i) - \kappa e^{-\frac{\alpha_i^2}{2\sigma^2}} \right)^2 \tag{2.2}$$

   Hint: You can use the numpy function `numpy.std` to get an estimate of the standard deviation of the random samples $\{x(k, \lambda)\}$. Together with the number of elements in the largest bin, you can use this to derive initial estimates of κ and σ in the Gaussian function.
   Your answer should consist of two parts:
   (1) Python script reading the random samples and the samples $h(\alpha_i)$ of the histograms $\boxed{\text{1 point}}$
   (2) 9 plots of the histograms and the Gaussian fits in one figure. Make sure the figure has no overlapping text! Comment on your results. $\boxed{\text{1 point}}$

8. Summarize the results Question 7 in the following table. Your answer should contain 3 tables (for different values of k) in the style of the table template in Table 2.2.

Table 2.2: The results of Gaussian fits for the random samples $\{x(k, \lambda)\}$ for $k = 10^3, 10^4, 10^5$ and $L = 100, 500, 2500$

L \ Parameter	$\hat{\kappa}$	$\hat{\sigma}$
50		
500		
5000		

$\boxed{\text{1 point}}$

9. Analyze how the standard deviation of the realizations and Gaussian fits is related to time and to the amount of realizations. Plot $\hat{\sigma}$ as a function of k (i.e. $N = 10^5$ time steps). How does $\hat{\sigma}$ depend on k? And how does the Gaussian fit change with respect to k and L? 3 points

10. Analyze the autocorrelation of the process. What is the relationship between the standard deviation $\hat{\sigma}(k)$, autocorrelation $r_x(k,k)$ and auto-covariance $c_x(k,k)$? Now, calculate the autocorrelation $r_x(k,k{+}500)$ for $k = \{0,1,2,\ldots 10^5 - 501\}$ using $L = 2500$ realizations. Plot the results. How does $r_x(k,k+500)$ depend on k? 2 points

11. Using the results obtained so far, what can you conclude regarding wide sense stationarity (WSS) of the random process? 1 point

2.3 Deliverables

You are required to hand-in a report, showing your results and answering the questions about this exercise. Python scripts should be included as text in your report. Do not forget to change the amount of interacting stars according to your student number, as no points can be awarded otherwise.

To help you with reporting, we provided a Jupyter notebook on Brightspace, which you can use as your report. We strongly encourage using the notebook, as hints and code snippets are already provided for you. After you have answered the questions and are finished with the assignment, you can export the Jupyter notebook as a PDF (instructions on how to do so are included in the notebook).

The report or exported Jupyter notebook should be dated and handed in before the deadline (see Course Schedule on Brightspace). The report should be handed in digitally by only one group member, but make sure the names and student numbers of all group members are included in the report or exported Jupyter notebook.

Bibliography

[1] K.J. Åström, *Noise and Control Design*, Excerpt from book manuscript 'Control of Microsystems', 2010.

[2] Robert Brown, *A brief account of microscopical observations made in the months of June, July and August, 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies*. Philosophical Magazine, Vol. 4, pp. 161–173, 1828.

[3] P. Chatterjee, L. Hernquist and A. Loeb, *Dynamics of a Massive Black Hole at the Center of a Dense Stellar System*, The Astrophysical Journal, Vol. 572(1), pp. 371–381, 2002.

[4] M. C. Wang, G. E. Uhlenbeck, *On the Theory of Brownian Motion II* Rev. Mod. Phys., Vol. 17(2-3), pp. 323–342, 1945.

[5] Paul Langevin, *Sur la théorie du mouvement brownien*, C. R. Acad. Sci. Paris, Vo. 146, pp. 530–533, 1908.

[6] G. Volpe and G. Volpe, *Simulation of a Brownian particle in an optical trap*, American J. Phys, Vol. 81(3), pp. 224–230, 2013.

[7] M. Hayes, *Statistical Digital Signal Processing and Modeling*, John Wiley & Sons, New York, 1996.

[8] M. Verhaegen, *Stochastic Processes for Physics Students: An Introduction*, TU Delft, 2019