IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 21, Issue 6, Ser. 3 (Nov. – Dec. 2025), PP 01-15
www.iosrjournals.org

Using The Chapman-Kolmogorov Equation For The Inference Of Discrete Random Walks

Mario Annunziato
Dipartimento Di Fisica "E. R. Caianiello", Università Degli Studi Di Salerno, Via G. Paolo II 132, 84084 Fisciano, Italy;

Abstract:
The the problem of inferring the jump probabilities of a discrete Random Walk from the probability distribution of the process out equilibrium is addressed. Numerical methods and algorithms for solving this problem in 1 and 2 dimensions, based on the Chapman-Kolmogorov equation, are investigated. The quantification of the a posteriori error of the calculated value is performed. Numerical experiments show the ability of the proposed computational procedure to infer the Random Walk parameters and the related accuracies.

Keywords: *Random Walk, inference, reconstruction, out equilibrium, Chapman-Kolmogorov.*

Date of Submission: 05-12-2025 Date of Acceptance: 15-12-2025

I. Introduction

The discrete Random Walk (RW) is the simplest Markov chain used for modelling the motion of non-deterministic time dependent observables. Discrete Markov chains are characterized from stochastic transition matrices [10] that are candidates for phenomenological modelling of observables for practical applications. Although modelling with Markov chains for discrete time and space domains could seem an over simplification of real random fluctuating phenomena, in some context they are the preferred tools of investigation, see e.g. [9, 18, 19]. Therefore, the estimation of the values of the transition matrix from measured observables plays a fundamental role in modelling. In statistical disciplines this task is called *inference*, in other research branches the terms *calibration*, i*dentification* or *reconstruction* are used.

The inference of the transition probability matrix of a Markov chain from sequences of sampled values dates back to the 1950s [1, 4, 6] and nowadays the research is still active. Following this early-stage research line, much of the current scientific literature is devoted on improvements of maximum likelihood statistical estimators from a single long sequence of data; see [3, 11, 13, 14, 22, 21] for a sample of literature. Within this, most of the analysis is focused on reversible Markov chains at equilibrium [13, 14, 22, 21] using the detailed balance condition. Some latest literature is concerned with the reconstruction from ensembles or aggregate data [5, 16, 17], when the single sequence is not available.

In this paper the analysis is centered on the inference of the inhomogeneous RW from ensemble data, i.e. from the probability distribution, evolving out of equilibrium or during transients, possibly with short observation time. In order to accomplish this task we use the feature of Markov processes, that is the Chapman-Kolmogorov equation, proposed in [2] for the identification of the Fokker-Planck equation. Moreover, we perform estimations of the *a posteriori* error for each single calculated value, that depend considerably on the numerical and measurement approximation errors.

The paper is organized as follows. In Section II we prepare the discussion with the definitions of inhomo- geneous RWs in 1 and 2 dimensions. In Sect. III the proposed methods and algorithms for the inference or reconstruction of the RW parameters and the related uncertainty estimates are illustrated. Numerical tests are provided in Sect. IV. Finally, a section of conclusions completes the paper.

II. Discrete space inhomogeneous random walk with drift

Following the same arguments of [2], our discussion is based on the inhomogeneous Random Walk (RW), also known as Bernoulli process, on the uniformly spaced discrete domain $\mathbb{M}_{\Delta x} \equiv \{x_m = m\Delta x, m \in \mathbb{Z}\}$ of size $\Delta x > 0$. If the walk position X is initially placed to $x^{(0)} \in \mathbb{M}_{\Delta x}$ and, at each time step i, it is subject to a displacement according to the outcomes of independent random jumps W_i, then at time n

$$X_n = x^{(0)} + \sum_{i=1}^{n} W_i, \tag{1}$$

where W_i may assume the values Δx or $-\Delta x$, outcoming with probability $p(x) \in [0,1]$ and $q(x) \in [0,1]$ respectively and such that $p(x) + q(x) = 1$. We define the drift $\mu(x)$ as

$$\mu(x) := p(x) - q(x) = 2p(x) - 1, \tag{2}$$

hence $p(x) = (1+\mu(x))/2, q(x) = (1-\mu(x))/2$.

The probability measure associated to this stochastic process X_n can be written as the sum of each path of the binary tree as follows

$$\mathbb{P}\{X_n = (2k-n)\Delta x + x^{(0)} | X_0 = x^{(0)}\} = \frac{1}{2^n} \sum_{\mathbf{z} \in C_2(n,k)} \prod_{i=0}^{n-1} 1 + \sigma(z_i)\, \mu(x^{(i)})\Big|_{x^{(i)} = x^{(0)} + \Delta x \sum_{j=1}^{i} \sigma(z_j)}. \tag{3}$$

This is the probability to find the process X_n at the position $x_m = x_{2k-n} + x^{(0)} \in \mathbb{M}_{\Delta x}$, where k is the number of positive Δx jumps of $W_i, i = 1, \ldots, n$, when it starts at $x^{(0)}$. This means that both k and n determine the displacement from the initial position according to the amount $(2k-n)\Delta x$. The symbol $\mathbf{z}$ represents a binary vector that spans over all the possible combination $C_2(n,k)$ of length n with k digits equal to 1, e.g. for $n = 4$ and $k = 2$ is $C_2(4,2) \equiv \{1100, 1010, 1001, 0110, 0101, 0011\}$, and $\sigma(z)$ is a sign function that is 1 for $z = 1$ and -1 for $z = 0$. The product represents the product of probabilities corresponding to the events of positive and negative jumps respectively.

Notice that the mean and the variance of X_n can not be written in a simple form as in the homogeneous case, because the random variables are not identically distributed at each time step n. From (3) it would be possible to write a combinatorial formula for the mean of X_n, but for our purposes this is not necessary. Indeed, we focus our discussion on the local evolution of the process, hence within this view the conditional mean $\mathbb{E}[X_{n+1} | X_n = x_m] = \mu_m \Delta x$ represents the local drift of the RW.

The further step of our discussion is to describe the 2-dimensional RW motion on a mesh [2]. Let $\mathbb{M}^2_{\Delta x, \Delta y} \equiv \{(x_{m_1} = m_1 \Delta x, y_{m_2} = m_2 \Delta y), \text{for } (m_1, m_2) \in \mathbb{Z}^2\}$ be the 2-dimensional discrete rectangular grid of step sizes Δx and Δy for each dimension. Likewise to the former definition, we write $\mathbf{X}_n = (X_n, Y_n)$ for the position at time n of the 2D RW with values on $\mathbf{x} = (x,y) \in \mathbb{M}^2_{\Delta x, \Delta y}$, and $\Delta\mathbf{x} = (\Delta x, \Delta y)$ for grid step sizes in vector form, where we use the bold characters for vectors symbols. At each time step the process initially placed in (x_{m_1}, y_{m_2}) moves towards one of the 4 nearest neighbour points of the grid according to the following random vector valued variable $\boldsymbol{\sigma}(\mathbf{x})$ defined in the following table

$$\boldsymbol{\sigma}(\mathbf{x}) = \begin{cases} (+1,0) & \text{with prob. } p_1(\mathbf{x}) \\ (0,-1) & \text{with prob. } p_2(\mathbf{x}) \\ (-1,0) & \text{with prob. } p_3(\mathbf{x}) \\ (0,+1) & \text{with prob. } p_0(\mathbf{x}), \end{cases} \tag{4}$$

jointly to the normalizing condition

$$p_0(\mathbf{x}) + p_1(\mathbf{x}) + p_2(\mathbf{x}) + p_3(\mathbf{x}) = 1. \tag{5}$$

The dependence of the jump probability on the grid position, defines an inhomogeneous RW. The 2D walk process

at the time step n, started at $\mathbf{X}_0 = \mathbf{x}^{(0)}$, reads as

$$\mathbf{X}_n = \mathbf{x}^{(0)} + \Delta\mathbf{x} \circ \sum_{j=1}^{n} \boldsymbol{\sigma}(\mathbf{X}_{j-1}), \tag{6}$$

where $\circ$ is the Hadamard product. For ease notation, in place of $\boldsymbol{\sigma}(\mathbf{x})$ we shall use the discrete random variable $z(\mathbf{x}) \in \{0,1,2,3\}$ to refer to the 4 possible outcomes with the corresponding probabilities $p_z(\mathbf{x})$. E.g. let $z = 3$ then according to (4) the outcome of $\boldsymbol{\sigma}$ is $\boldsymbol{\sigma}(z) = (-1,0)$. The definition of the RW can be given in terms of the drifts $\mu_1(x,y)$, $\mu_2(x,y)$ along the two dimensions and a function $a(x,y)$, according to the following relationship

$$\begin{aligned} p_1(x,y) &= \frac{1}{2}(a(x,y) + \mu_1(x,y)) \\ p_3(x,y) &= \frac{1}{2}(a(x,y) - \mu_1(x,y)) \\ p_2(x,y) &= \frac{1}{2}(-a(x,y) - \mu_2(x,y) + 1) \\ p_0(x,y) &= \frac{1}{2}(-a(x,y) + \mu_2(x,y) + 1). \end{aligned} \tag{7}$$

The probability measure is defined as follows. Let k_1 be the number of positive jumps along the x direction, n_1 be the total number of jumps along the x direction, k_0 be the number of positive jumps along the y direction and n_2 be the total number of jumps along the y direction, then for (6) the conditional probability to find the process at time $n \geq 1$ to the point $(x_m, y_l) = (x_{2k_1-n_1} + x^{(0)}, y_{2k_0-n_2} + y^{(0)})$ is

$$\begin{aligned} \mathbb{P}\{\mathbf{X}_n = &((2k_1 - n_1)\Delta x, (2k_0 - n_2)\Delta y) + \mathbf{x}^{(0)} | \mathbf{X}_0 = \mathbf{x}^{(0)}\} = \\ &\sum_{\mathbf{z}=C_4(n,n_1,n_2,k_1,k_0)} \prod_{i=0}^{n-1} p_{z_i}(\mathbf{x}^{(i)}) \Bigg|_{\mathbf{x}^{(i)} = \mathbf{x}^{(0)} + \Delta\mathbf{x} \circ \sum_{j=1}^{i} \sigma(z_j)}, \end{aligned} \tag{8}$$

where $\mathbf{z} = (z_0, z_1, \ldots, z_n)$ is a vector whose elements span over the set of all $C_4(\ldots)$ combinations of the digits $0,1,2,3$ of length n, with k_0 the count of $0's$, k_1 the count of $1's$, n_1 is the count of the odd numbers 1 and 3, n_2 the count of the even numbers 0 and 2, such that $n = n_1 + n_2$, and $k_1 \leq n_1$, $k_0 \leq n_2$. As an example, for $\mathbf{z} = 30110231$ it is $n = 8$, $n_1 = 5$, $n_2 = 3$, $k_0 = 2$, $k_1 = 3$.

When the random walk is homogeneous, i.e. p_z are independent on $\mathbf{x}$, the formula for the probability measure is explicit with Newton's binomials terms as follows

$$\begin{aligned} \mathbb{P}\{\mathbf{X}_n = &((2k_1 - n_1)\Delta x, (2k_0 - n_2)\Delta y) + \mathbf{x}^{(0)} | \mathbf{X}_0 = \mathbf{x}^{(0)}\} = \\ &\sum_{n_1=0}^{n} \binom{n}{k_1} \binom{n-k_1}{n_1-k_1} \binom{n-n_1}{k_0} p_1^{k_1} p_2^{k_2} p_3^{k_3} p_0^{k_0}. \end{aligned} \tag{9}$$

Finally, we notice the difference in the mapping of coordinates between the 1D and 2D cases. From Eq. (3) for a given time step n and the number of positive jumps k, the coordinate of the displacement from the initial position is uniquely determined. Instead, from Eq. (8) for a given time step n and positive jumps k_0, k_1 in the two dimensions, the displacement from the initial position is not uniquely determined, since there is one degree of freedom in the choice of n_1 and n_2 set by $n = n_1 + n_2$.

Notice that these RWs can be defined in terms of stochastic transition matrices. We call *direct* problems when these matrices are given and the Markov chain properties are analysed. For instance, given the drift at each grid points $\mu(x_m)$, $x_m \in \mathbb{M}_{\Delta x}$ and the initial distribution f_m^0 at time $n = 0$, by performing Monte Carlo simulations of the RW we can obtain the experimental distribution f_m^n of the probabilities of the process at each time step n, that will be used for the purpose of numerical experiments.

In general the RW is *dispersive* with finite velocity of propagation, that is if we set the initial value of the RW concentrated at the point $x^{(0)}$, then at the time step n the distribution of the process will be contained into

the discrete domain $[x^{(0)} - n\Delta x, x^{(0)} + n\Delta x] \subset \mathbb{M}_{\Delta x}$, whereas outside this domain the distribution will be surely vanishing. Aside special values of $\mu(x)$ this process has no equilibrium and without domain boundaries.

II.1 Error estimates

We recall here the basic statistical error estimates of the distribution generated from a RW, that will be used in the section of numerical experiments. Let R be the number of repetitions of an experiment for the estimation of the expected value $\bar{f}$ of the measured probability f_m^n of the random walk at the location x_m at time n, and ξ_i the binary random variable whose value is 1 if $X_n = x_m$ and 0 if $X_n \neq x_m$.

For gaussian random variables we use a Chebyshev-like inequality [12] for large R

$$\mathbb{P}(|\sum_{i=1}^{R} \xi_i/R - \bar{f}| \geq \varepsilon) \leq \frac{\sigma^2}{3R\varepsilon^2}. \tag{10}$$

Thus, if we settle on the probability that the frequency approximates the expected value with an approximation value ε, that is $R\varepsilon^2$ is constant, we see that as ε goes to zero the number of Monte Carlo simulation must grow as R^2. More precisely, let $\alpha = \sigma^2/(3R\varepsilon^2)$ be the *maximum deviation probability* of the confidence interval ε, then it can be estimated as $\varepsilon = \sqrt{\sigma^2/(3\alpha R)}$. Furthermore if the values of f_m^n are close to 0, then $\sigma^2 \simeq \bar{f}$, so that

$$\varepsilon = \sqrt{\frac{\bar{f}}{3\alpha R}}. \tag{11}$$

Notice that the Bernstein inequality [23] used for limited random variables gives little improvements.

In the following in place of the maximum deviation probability we shall use the *significance level*, albeit this definition is not properly correct in this context. As an example, if we suppose for f an uniform distribution on the domain of 40 points, we get $\bar{f} \approx 0.025$. Let $R = 10^6$ and $\alpha = 0.2$ is $\varepsilon_a(f) \approx 2 \cdot 10^{-4}$, equivalent to relative error $8 \cdot 10^{-3}$ for f.

III. The reconstruction of the RW parameters

In the previous section we have introduced the inhomogeneous Random Walk process X_n, as the random movement of a particle on a grid, that is driven according to the jump probabilities depending on the position on the grid. The inverse problem faces the reconstruction or the inference of the jump probabilities, i.e. the drifts, at each grid point, when the probability distribution of X_n is known.

III.1 One dimensional case

Our framework for solving this problem is based on the Chapman-Kolmogorov equation for the RW process. Let f_m^n be the probability that the particle is located at x_m at the time step n, we have the following CK equation

$$f_m^{n+1} = f_{m+1}^n(1 - p_{m+1}) + f_{m-1}^n p_{m-1} \tag{12}$$

stating that the probability f_m^{n+1} to find the particle in m at the time step $n+1$ is the sum of the two probabilities arriving from the neighbours, each calculated as the product of the probability at the previous time step by the jump probability (left or right) at that locations. For the inverse problem the f_m^n are supposed given and p_m are unknowns.

As we have discussed in the former section, when $p \in (0,1)$ the RW represents a dispersive dynamics, meaning that at each time step the domain of the process increases of 2 points, in this sense we refer to it as a moving domain. Whereas for the cases $p = 0$ and $p = 1$ the process is deterministic with a preferred direction, hence it can no longer be called dispersive. From now onwards we suppose $p \in (0,1)$ except eventually at the boundary

of the domain, as we shall discuss. Outside this domain the probability of the process is zero, hence we use this information to set the boundary values for the system (12), as follows. Let us suppose that at time $n+1$ the domain of the probability distribution, where it is surely non-vanishing is indexed by $m = 1,\dots M$, and for $m \leq 0$ and $m \geq M+1$ the probabilities f_m^{n+1} will be surely zero, then from Eq. (12) we see that only $M-2$ unknowns p_m can be determined, for $m = 2,\dots,M-1$. Furthermore, at the time step n the domain has size $M-2$ for $m = 2,\dots M-1$ and will be surely $f_1^n = f_M^n = 0$. This missing of information is intrinsic into the nature of the Chapman-Kolmogorov equation and, unless the boundaries values of p are known at the boundary of the domain, the values p_1 and p_M can not be determined.

Therefore, from this discussion we have found that the Eq. (12) must be completed with the boundary condition $f_0^n = f_1^n = f_M^n = f_{M+1}^n = 0$. The values of p_m can be found from Eq. (12) by recursion, provided that the starting values are $p_2 = (f_2^n - f_1^{n+1})/f_2^n$ and $p_3 = (f_3^n - f_2^{n+1})/f_3^n$.

Instead, for a static bounded domain a reflecting barrier [8] can be placed at the boundary $M = 1$, that is $p_1 = 1$ is known, hence we get $p_3 = (f_3^n + f_1^n - f_2^{n+1})/f_3^n$ because it is supposed that f_1^n may be non-vanishing. We will use the last expression for p_3, which accounts for the case $f_1^n > 0$, in such a way that when it is non-negative, we set $p_1 = 1$, i.e. a reflecting barrier at the boundary domain $m = 1$, whereas in the case $f_1^n = 0$ we say the process has not any barrier or the barrier has not yet been reached at the domain's boundary so that the value of p_1 can not be determined. Similar remarks can be done at the other boundary $m = M$.

With this initial preparation, from Eq. (12) starting with $m = 2$ and by recursion, we get the following (forward) explicit formula for p_m

$$p_m = \frac{\delta_{m,3} f_1^n + \sum_{j=0}^{\lfloor (m-2)/2 \rfloor} f_{m-2j}^n - f_{m-1-2j}^{n+1}}{f_m^n} \qquad m = 2,\dots,M-1, \tag{13}$$

where the symbol $\lfloor m \rfloor$ is the greatest integer less than m and $\delta_{m,3}$ is the Kronecker-δ symbol for $m = 3$. For the drift we have $\mu_m = 2p_m - 1$. Notice that this formula expresses the impossibility to calculate the value of p_m for a vanishing values of f_m^n. A similar explicit formula can be derived from (12) by starting the computation from $m = M-1$ and proceeding the recursion with backward counting of m.

The p_m computed with (13) are subject to the accuracy of f_m^n as in the case of real observables measurements or statistical fluctuations. This influence could be so strong that the calculated values of p_m could lay outside the theoretical interval $[0,1]$. For this reason a quantification of the uncertainty of the solution is an important task of the reconstruction problem [15]. The standard law of error propagation is a basic approach to achieve this task. Let $\varepsilon_a(f_m^n)$ be the absolute error associated to f_m^n, then from (13) the absolute error $\varepsilon_a(\mu_m)$ for the calculated drift μ_m reads as

$$\varepsilon_a(\mu_m) = \frac{2}{f_m^n}\left(|p_m| \varepsilon_a(f_m^n) + \delta_{m,3}\varepsilon_a(f_1^n) + \sum_{j=0}^{\lfloor (m-2)/2 \rfloor} \varepsilon_a(f_{m-2j}^n) + \varepsilon_a(f_{m-1-2j}^{n+1}) \right). \tag{14}$$

Notice that this formula allows to estimate the error of the drift punctually, i.e. for each point of the grid.

The algorithm for (13) and (14), and its backward version, can be formulated as follows.

Algorithm 1

1. *Input data: probability distributions f_m^n and f_m^{n+1} and related absolute errors $\varepsilon_a(f_m^n)$ and $\varepsilon_a(f_m^{n+1})$ of the RW at the 2 time steps n and $n+1$ for $m = 1,\dots,M$; computation direction flag $dc = 1$ for forward, $dc = 0$ for backward.*
2. *Initial settings: set the unknowns values $p_m = NaN$ and errors $\varepsilon_a(\mu_m) = NaN$, for $m = 1,\dots,M$.*
3. *if $dc == 1$ continue. if $dc == 0$ jump to step 8.*
4. *If $f_1^n > 0$ recognize a reflecting barrier at $m = 1$ and set $p_1 = 1$.*

5. *Starting values:* $p_2 = f_2^n - f_1^{n+1}$, $p_3 = f_3^n - f_2^{n+1} + f_1^n$; $\varepsilon_a(\mu_2) = \varepsilon_a(f_2^n) + \varepsilon_a(f_1^{n+1})$, $\varepsilon_a(\mu_3) = \varepsilon_a(f_3^n) + \varepsilon_a(f_3^{n+1}) + \varepsilon_a(f_1^n)$.
6. *Iteration formula: for* $m = 3$ *to* $M-2$, *do* $p_{m+1} = f_{m+1}^n - f_m^{n+1} + p_{m-1}$ *and* $\varepsilon_a(\mu_{k+1}) = \varepsilon_a(f_{k+1}^n) + \varepsilon_a(f_k^{n+1}) + \varepsilon_a(\mu_{k-1})$, *end-for.*
7. *Jump to step 11.*
8. *If* $f_M^n > 0$ *recognize a reflecting barrier at* $m = M$ *and set* $p_M = 0$.
9. *Starting values:* $p_{M-1} = f_M^{n+1}$, $p_{M-2} = f_{M-1}^{n+1} - f_M^n$; $\varepsilon_a(\mu_{M-1}) = \varepsilon_a(f_M^{n+1})$, $\varepsilon_a(f_{M-2}) = \varepsilon_a(f_{M-1}^{n+1}) + \varepsilon_a(f_M^n)$.
10. *Iteration formula: for* $m = M-2$ *to* 3, *do* $p_{m-1} = -f_{m+1}^n + f_m^{n+1} + p_{m+1}$ *and* $\varepsilon_a(\mu_{m-1}) = \varepsilon_a(f_{m+1}^n) + \varepsilon_a(f_m^{n+1}) + \varepsilon_a(\mu_{m+1})$, *end-for.*
11. *Extract the probabilities and the errors: for* $m = 2$ *to* $M-1$, *do* $p_m = p_m / f_m^n$ *and* $\varepsilon_a(\mu_m) = (\varepsilon_a(\mu_m) + \varepsilon_a(f_m^n)|p_m|)/f_m^n$, *end-for;*
12. *Find the local drift:* $\mu_m = 2 * p_m - 1$ *and its related absolute error* $\varepsilon_a(\mu_m) = 2\varepsilon_a(\mu_m)$.

Notice that when $f_m^n = 0$ *the corresponding* μ_m *is set to NaN.*

We have included the case of a reflecting barrier for a finite domain, although it will not be used in the numerical experiments.

Everything illustrated so far is the Algorithm that by using two time steps of the probability distribution of the RW, calculates the unknown drifts by using all the space points at once. Notice that this algorithm can not be extended to the 2 dimensional case on the square grid.

The other option for the drift estimation is the using of the CK Eq. (12) for a given m evaluated at two different time steps n_1, n_2 in order to build a system of 2 equations for the unknowns μ_{m-1}, μ_{m+1}, rather than p_m, as follows

$$\begin{pmatrix} -f_{m-1}^{n_1} & f_{m+1}^{n_1} \\ -f_{m-1}^{n_2} & f_{m+1}^{n_2} \end{pmatrix} \begin{pmatrix} \mu_{m-1} \\ \mu_{m+1} \end{pmatrix} = \begin{pmatrix} -2f_m^{n_1+1} + f_{m+1}^{n_1} + f_{m-1}^{n_1} \\ -2f_m^{n_2+1} + f_{m+1}^{n_2} + f_{m-1}^{n_2} \end{pmatrix} \tag{15}$$

then solving it for each $m = 2, \ldots, M-1$. This method is not equivalent to the previous since the μ_m are calculated locally in space from a system of only 2 equations. From numerical experiments we found that this method is very sensitive to uncertainties and it is vulnerable to singularity or ill-conditioness of the coefficient matrix. Therefore, in order to improve the accuracy of the solution, we consider the CK equation at $n_1, \ldots, n_K$ time steps in order to increase the number of equations in the system, so that (15) becomes overdetermined with the coefficient matrix rectangular of size $(K, 2)$, as follows

$$\begin{pmatrix} -f_{m-1}^{n_1} & f_{m+1}^{n_1} \\ -f_{m-1}^{n_2} & f_{m+1}^{n_2} \\ \cdots & \cdots \\ -f_{m-1}^{n_K} & f_{m+1}^{n_K} \end{pmatrix} \begin{pmatrix} \mu_{m-1} \\ \mu_{m+1} \end{pmatrix} = \begin{pmatrix} -2f_m^{n_1+1} + f_{m+1}^{n_1} + f_{m-1}^{n_1} \\ -2f_m^{n_2+1} + f_{m+1}^{n_2} + f_{m-1}^{n_2} \\ \cdots \\ -2f_m^{n_K+1} + f_{m+1}^{n_K} + f_{m-1}^{n_K} \end{pmatrix}. \tag{16}$$

This system in the form $\hat{A}\mu = \hat{b}$ can be solved as a least squares problem

$$\min_{\mu} \|\hat{A}\mu - \hat{b}\|_2, \quad \text{i.e. } \hat{A}^T\hat{A}\mu = \hat{A}^T\hat{b}, \tag{17}$$

by using the Moore-Penrose pseudoinverse of the coefficient matrix. With the purpose to improve the quality of the numerical results, we add a Tichonov term of regularization

$$\min_{\mu} \frac{1}{2}(\|\hat{A}\mu - \hat{b}\|_2^2 + \gamma\|Q\mu - \beta\|_2^2), \tag{18}$$

where γ is the regularization weight, Q is a vector or a matrix and correspondingly β is a scalar or a vector, both encoding the type of regularization on the unknown p. The solution of this minimization problem is achieved by solving the following system

$$\begin{bmatrix} \hat{A} \\ \gamma Q \end{bmatrix} \mu = \begin{bmatrix} \hat{b} \\ \gamma \beta \end{bmatrix}. \tag{19}$$

The choice

$$Q = (1, -1) \tag{20}$$

and $\beta = 0$, minimize the derivative of the drift μ.

For the error estimation to the solution of this problem, when the data are affected from uncertainty, we follow the valuable analysis of [7] providing the component wise error estimation to the solution x of a full rank equation system $Ax = b$, when A and b are subject to perturbations. For convenience, we report here that useful results.

Theorem 1 ([7]) *Let $Ax = b$ be an equation system with full rank matrix A and δA and δb be some perturbations of the system matrix and known term respectively, ω_A and ω_b be the relative errors defined by the component wise inequalities as follows: $|\delta A_{ij}| \leq \omega_A |A_{ij}|$, $|\delta b_i| \leq \omega_b |b_i|$, and $P_{\mathcal{N}(A^T)} = I - AA^\dagger$, then provided that the spectral radius ρ of the matrix*

$$\mathscr{A} := \begin{pmatrix} |A^\dagger|^T |A|^T & |P_{\mathcal{N}(A^T)}||A| \\ |(A^T A)^{-1}||A|^T & |A^\dagger||A| \end{pmatrix} \tag{21}$$

satisfy $\rho(\mathscr{A}) \leq 1/\omega$, with $\omega = \max(\omega_A, \omega_b)$, then the absolute error on the solution x is bounded by the following component wise inequality

$$|\delta x| \leq \omega \Big(|A^\dagger|(|b| + |A||x|) + |(A^T A)^{-1}||A|^T |r| \Big) + O(\omega^2), \tag{22}$$

where $r = b - Ax$ is the residual of the equation system and $A^\dagger$ denotes the Moore-Penrose pseudoinverse.

The expression inside the brackets provides an estimate of the amplification of the data uncertainty. Notice that for the numerical implementation of (22) a better choice is to check for the full rank of $A^T A$ rather than A.

This important inequality provides estimates of uncertainties of the calculated drifts (or probabilities) μ_i for each grid point of the random walk domain, provided the setting $A := \hat{A}^T \hat{A}$ and $b := \hat{A}^T \hat{b}$.

By summarizing, if the maximum relative errors ω of the input data f is known, then (22) provides component wise error estimates for each component the unknown μ. The algorithm for computing this reconstruction problem can be formulated as follows.

Algorithm 2

1. *Input data: probability distribution f_m^n of the RW for $m = 1 \ldots M$ and for $n = n_1, \ldots, n_K$ time steps, the weight γ and the regularization parameters Q and β of* (20)*, the maximum relative error ω for f_m^n. Set the vector μ_m for saving the results and $\delta\mu_m$ for the bounding errors. Set a vector counter c_m to zero.*
2. *Iterate over the space grid points $m = 2 \ldots M - 1$.*
3. *Build the matrices of Eq.* (16) *for a given m and all n, then the system* (19)*.*
4. *If the system matrix has not rank* 2*, then jump the next two steps.*
5. *Solve the system of Eq.* (17) *with the pseudoinverse method. Store the two solutions temporary in $\tilde{\mu}_{-1}$ and $\tilde{\mu}_{+1}$. Calculate the component wise bounding errors $\delta\tilde{\mu}_{-1}, \delta\tilde{\mu}_{+1}$ with* (22)*.*
6. *Check the counters c_i for $i = m-1, m+1$: If $c_i == 0$ then $\mu_i = \tilde{\mu}_{i-m}$, elseif $c_i > 0$ then $\mu_i = \mu_i + \tilde{\mu}_{i-m}$. Do the same for the bounding errors $\delta\mu_i$. Increment the counters by 1: $c_i = c_i + 1$.*

7. *End of iteration 2.*

8. *Divide μ_m and $\delta\mu_m$ for the value of the counters c_m. Now μ_m and $\delta\mu_m$ are calculated. End.*

Notice that the counter c_m keeps track of the of the calculated μ_m, since it could be calculated twice. At the step 5. the Matlab®[20] functions `lsqminnorm` or `lsqnonneg` provide a better numerically stable routines for solving the equation system.

III.2 Two dimensional case

This section is devoted to the solution of the inverse problem for the the 2D-RW (6), i.e. the calculation of the table (4) for each point $\mathbf{x}$ of the domain when the 2D probability distribution $f^n_{m,l}$ is given on $\mathbb{M}^2_{\Delta x,\Delta y}$ for some values of n. To pursue this aim we follow the technique illustrated in [2]. For convenience we report here the definitions.

Let $p^{(s)}_{m,l} = p_s(x_m, y_l)$ the unknowns probabilities, the key relationship with $f^n_{m,l}$ is established by the 2-dimensional CK equation analogous to Eq. (12), that is

$$f^{n+1}_{m,l} = f^n_{m-1,l}p^{(1)}_{m-1,l} + f^n_{m,l+1}p^{(2)}_{m,l+1} + f^n_{m+1,l}p^{(3)}_{m+1,l} + f^n_{m,l-1}p^{(0)}_{m,l-1}. \tag{23}$$

This equation conveys the probability that the RW reaches the point (m,l) at time $n+1$ by jumping from the 4 nearest neighbours, according to the table (4). The total probability distribution is subject to conservativeness according to $\sum_{m,l} f^n_{m,l} = 1,\ \forall n$.

By using many CK equations (23), for $n = n_1, \ldots, n_K$ with $K \geq 4$, we build the over-determined equation system

$$\begin{pmatrix} f^{n_1}_{m-1,l} & f^{n_1}_{m,l+1} & f^{n_1}_{m+1,l} & f^{n_1}_{m,l-1} \\ f^{n_2}_{m-1,l} & f^{n_2}_{m,l+1} & f^{n_2}_{m+1,l} & f^{n_2}_{m,l-1} \\ f^{n_3}_{m-1,l} & f^{n_3}_{m,l+1} & f^{n_3}_{m+1,l} & f^{n_3}_{m,l-1} \\ \cdots & \cdots & \cdots & \cdots \\ f^{n_K}_{m-1,l} & f^{n_K}_{m,l+1} & f^{n_K}_{m+1,l} & f^{n_K}_{m,l-1} \end{pmatrix} \begin{pmatrix} p^{(1)}_{m-1,l} \\ p^{(2)}_{m,l+1} \\ p^{(3)}_{m+1,l} \\ p^{(0)}_{m,l-1} \end{pmatrix} = \begin{pmatrix} f^{n_1+1}_{m,l} \\ f^{n_2+1}_{m,l} \\ f^{n_3+1}_{m,l} \\ \cdots \\ f^{n_K+1}_{m,l} \end{pmatrix}, \tag{24}$$

that can be solved as a least-square problem $\min_p \|Ap - b\|_2^2$, similarly to (17).

The algorithm implementing the solution of the inverse problem is based on Algorithm 1 of [2], as follows.

Algorithm 3

1. *Input data: probability distribution $f^n_{m,l}$ of the RW, for $n = n_1, n_2, \ldots, n_K$ time steps, the maximum relative error ω for f^n_m. Set the matrices $p^{(s)}_{m,l}$ and $\delta p^{(s)}_{m,l}$ to save the results and the bounding errors. Set* $\text{count\_p}^{(s)}_{m,l} = 0$ *matrix to flag the calculated values of $p^{(s)}_{m,l}$.*

2. *Start the iteration over all the values of (m,l) of the grid domain given by the size of $f^n_{m,l}$.*

3. *Build the matrices of Eq.* (24) *for a given (m,l) and for all $n = n_1, n_2, \ldots, n_K$.*

4. *If the system matrix has not rank* 4*, then jump the next step.*

5. *Solve the system* (19) *with the Moore-Penrose pseudoinverse. Calculate the component wise bounding errors $\delta p^{(s)}_{m,l}$ with* (22)*. Cumulate the calculated values of $p^{(s)}_{m,l}$ and increments the counter* $\text{count\_p}^{(s)}_{m,l}++$*. During the iteration over all points (m,l) of the domain, the $p^{(s)}_{m,l}$ could be calculated more than once repeatedly. The counters* $\text{count\_p}^{(s)}_{m,l}$ *keep track of the calculated values of the corresponding $p^{(s)}_{m,l}$.*

6. *End of iteration 2.*

7. *Averages the calculated p:* $p^{(s)}_{m,l}/\text{count\_p}^{(s)}_{m,l}$, *in order to take in account for the eventually repetitions of the calculated values of* $p^{(s)}_{m,l}$.

8. *With* $p^{(s)}_{m,l}$ *calculate the required* $\mu^{(1)}_{m,l}, \mu^{(2)}_{m,l}$ *and* $a_{m,l}$.

Notice that this algorithm does not guarantee the normalization of the jump probabilities (5).

IV. Numerical experiments

In this section we illustrate the numerical experiments for testing the algorithms for the reconstruction problems. The goal of our problem is the inference of the drift $\mu(x)$ of the RW from the measured experimental probability distribution f^n_m for a set of values of n. For the purpose of numerical tests, we set the object of calibration $\mu(x)$, then, in order to mimic the measured probability distribution f^n_m, starting from a random initial point with a given distribution, we computationally generate the values f^n_m with Monte Carlo bunches of the RW (1). The estimates $\mu_m \approx \mu(x_m)$ are found by using the proposed algorithms. Similar tests are performed for $\mu_1(x,y), \mu_2(x,y)$ and $a(x,y)$ in the 2D case.

IV.1 Numerical experiments for 1D-RW

For the first test we set the drift function

$$\mu(x) = -0.2(1+0.02x) + 0.03x\Theta(x-5) - 0.02x\Theta(-x-10) \tag{25}$$

with two discontinuities in $x = 5$ and $x = -10$, marked by the $\Theta(\cdot)$ step functions. Indeed, this function is evaluated on the discrete domain $\mathbb{M}_1$, hence continuity is meaningless in this context, but the choice is motivated for testing with a rapidly changing function.

The data of the RW are generated with a grid step size $\Delta x = 1$, starting at a point randomly chosen with uniform distribution on the grid points $[-4,4] \cap \mathbb{M}_1$, then evolving for $N = 15$ time steps, according to the jump probability $p(x)$ related to the drift (25) with (2), i.e. $p(x)$ and $q(x)$, evaluated on the discrete RW domain $[-19,19] \times [0,14] \cap \mathbb{M}_1 \times \mathbb{T}_1$.

The experimental discrete probability distribution f^n_m is calculated by recording the position of the RW in normalized histograms with a binning process, resulting from a bunch of MC simulations with R repetitions. For the purpose of the statistical convergence test, the bunches of size $R = 10^4, 10^5, 10^6, 10^7, 10^8$ are executed.

In Fig. 1 we show a sample of the generated f^n_m evolving during the transient represented by histograms, for a MC bunch with $R = 10^6$.

In Fig. 2 we show the results of the reconstruction with Alg. 1 for data f^n_m generated with $R = 10^7$, at time step $n = 14$. The same data are individually used in place of $\bar{f}$ in (14) with $\alpha = 0.2$ for the estimation of $\varepsilon_a(f^n_m)$. In particular, the left picture depicts the computed μ_m in the forward direction, dc=1, where the black dots represent the computed μ_m and the errorbars the estimated uncertainty intervals with a significance level $\alpha = 0.2$. The lines are the reference solution of the true $\mu(x)$. We note that the estimated errors are large at the boundary of the domain, where the values of the probabilities are closer to zero and more sensitive to statistical errors. This effect is more pronounced on the right side of the domain, because of the cumulative effect of the forward computation that proceed from left to right. The results of the backward computation are not shown, since they are similar, but with errorbars larger to the left side of the domain. In the right picture of Fig. 2 we show the results of the two computations merged in a such way to take, for each point of the domain, the value with the smaller errorbar. We can appreciate the good agreement with the given $\mu(x)$ within the estimated uncertainty.

In Fig. 3 we show the reconstructed $\mu(x)$ with Alg. 2 for data generated with $R = 10^8$ and $n = 2, \ldots, N-2$. Errorbars are calculated by using the Björk formula (22). Notice that for some points the errorbars were not

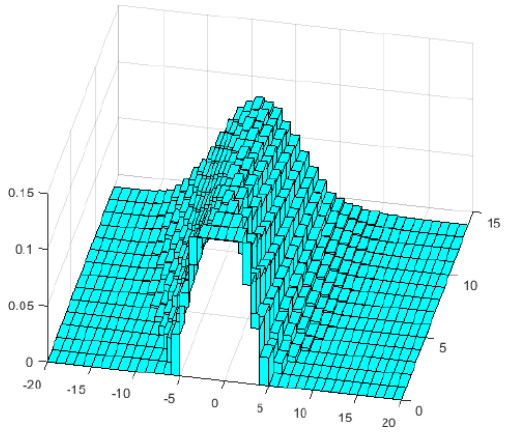

Figure 1: Histograms of the evolution of the computationally generated probability distribution of the RW with $R = 10^6$.

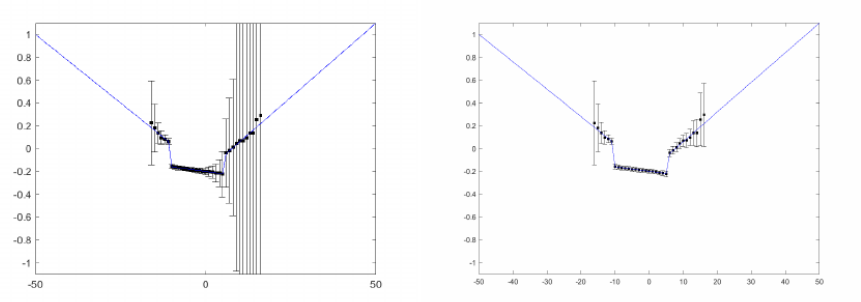

Figure 2: Computed μ_m with Alg. 1 and $R = 10^7$ vs. exact $\mu(x)$. Left: forward computation. Right: post-processed with merged forward-backward computation results.

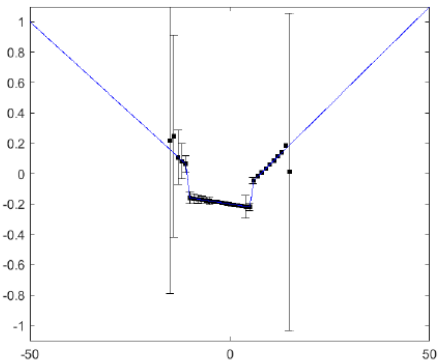

Figure 3: Computed μ_m with Alg. 2 vs. exact $\mu(x)$. $R = 10^8$.

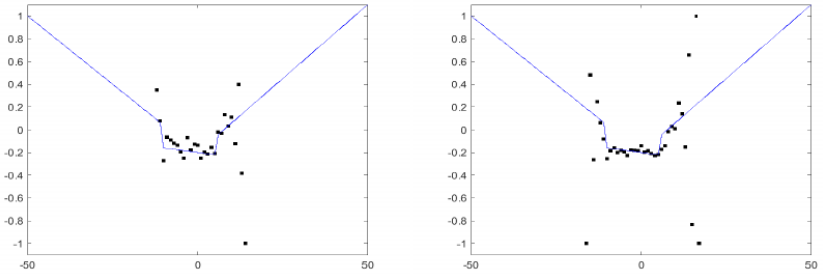

Figure 4: Action of the regularization in Alg. (2) on the computed drift, $R = 10^4$. Left without regularization. Right regularized with $\gamma = 1$.

Table 1: Convergence results as a function of the number of samples R

R	$\varepsilon_2(\mu)$ Alg. 1	$\varepsilon_2(\mu)$ Alg. 2
$10^4, \gamma = 1$	-	$6.80 \cdot 10^{-2}$
10^4	$6.54 \cdot 10^{-2}$	$2.88 \cdot 10^{-1}$
10^5	$2.05 \cdot 10^{-2}$	$1.02 \cdot 10^{-1}$
10^6	$1.10 \cdot 10^{-2}$	$1.60 \cdot 10^{-2}$
10^7	$3.60 \cdot 10^{-3}$	$1.25 \cdot 10^{-2}$
10^8	$9.73 \cdot 10^{-4}$	$7.04 \cdot 10^{-3}$

calculated because of the failure in satisfying the hypothesis of Thm 1. We see the good prediction, albeit with larger errorbars with respect to those estimated with Alg. 1.

Fig. 4 shows the calculated μ_m without regularization on the left, and with regularization on the right, with $\gamma = 1$, for f_m^n generated with $R = 10^4$.

For the convergence test, we define the average square error between the given and computed drift, as follows

$$\varepsilon_2(\mu) = \frac{\|\mu(\cdot) - \mu_\cdot\|_2}{\text{size}(\mu)}.$$

In Table 1 is reported the result of convergence test for different number of samples R of the RW for both algorithms. We see the error decreases proportionally to the square root $\sqrt{R}$. For the case with $R = 10^4$ the test with regularization shows a relevant improvement of the accuracy of μ_m. Anyway, for larger value $R \geq 10^5$ of samples this effect is lost.

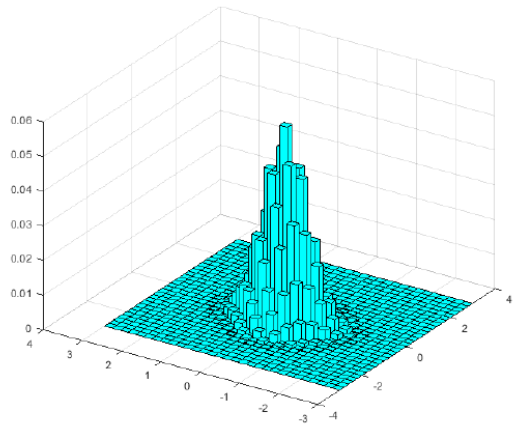

Figure 5: Probability distribution of the 2D RW at the final time $T = 14$.

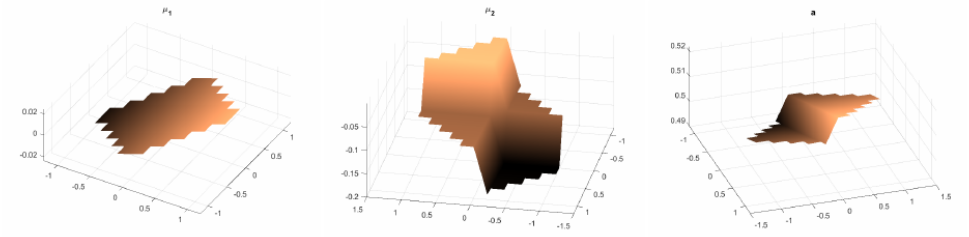

Figure 6: Computed μ_1, μ_2, a for the 2D Random Walk, portrayed as surfaces.

IV.2 Numerical experiments for 2D-RW

For the 2D RW we set the following objects of reconstruction

$$\begin{aligned} \mu_1(x,y) &= 0.02x \\ \mu_2(x,y) &= 0.01y - 0.1\Theta(y+0.1) - 0.1\Theta(-x) \\ a(x,y) &= 0.5 + 0.01x + 0.01\Theta(y+0.1), \end{aligned} \tag{26}$$

corresponding to

$$\begin{aligned} p_0(x) &= (0.5 - 0.01x + 0.01y - 0.11\Theta(y+0.1) - 0.1\Theta(-x))/2 \\ p_1(x) &= (0.5 + 0.03x + 0.01\Theta(y+0.1))/2 \\ p_2(x) &= (0.5 - 0.01x + 0.01\Theta(y+0.1))/2 \\ p_3(x) &= (0.5 - 0.01x - 0.01y + 0.09\Theta(y+0.1) + 0.1\Theta(-x))/2. \end{aligned}$$

Here we do not compute the probability distribution with MC simulations, rather with the exact formulas (8) and (7). This is motivated by the high sensitivity of the inverse problem that will be discussed below. The grid step sizes are $\Delta x = \Delta y = 0.1$. Fig. 5 depicts the distribution $f^n_{m,l}$ computed by using that exact formulas with (26), starting with probability 1 at the coordinate $(0,0)$, for $N = 14$ time steps on the domain $[-3,3] \times [0,14] \cap \mathbb{M}_{0.1,0.1} \times \mathbb{T}_1$.

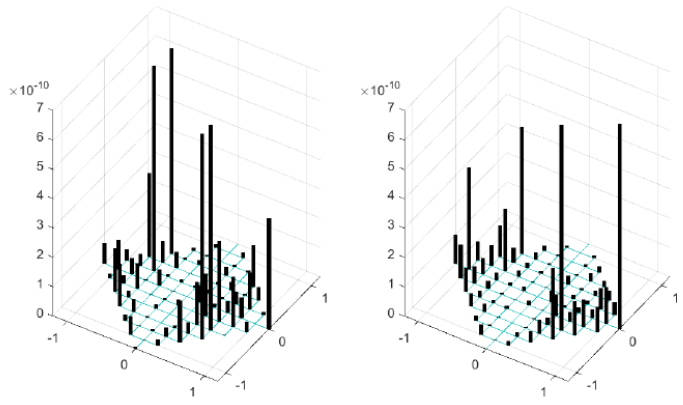

Figure 7: Left: absolute error between the computed and true μ_2. Right: absolute error estimate of the computed μ_2 calculated with a relative error of $3 \cdot 10^{-15}$ on the distribution.

$\varepsilon(\bar{f})$	true $\varepsilon_2(\mu_2)$	Björk $\varepsilon_2(\mu_2)$
$N\varepsilon_{FPU} = 3 \cdot 10^{-15}$	$4.07 \cdot 10^{-11}$	$1.45 \cdot 10^{-10}$
$10^{-7}, R = 20$	$1.23 \cdot 10^{-3}$	$3.69 \cdot 10^{-3}$

Table 2: Test for accuracy of the computed drift μ_2. True and estimated error with the Björk formula.

The input data $f^n_{m,l}$ for Alg. 3 are selected for $n = 2, \dots N$ and the relative accuracy is set $\varepsilon = N \cdot 10^{-16}$. This follows from (8), since the RW accumulates the floating point error ε_{FPU} at each step of the walk. Indeed, this accuracy assumption represents a lower bound, because this value should be multiplied by the Newton's binomial products. The objects of reconstruction a, μ_1 and μ_2 computed with Alg. 3 are shown in Fig. 6.

With the aim to check the algorithm with a lower data accuracy, we performed a MC second test as follows. A gaussian random variable with a standard deviation 10^{-7} was added to the exact distribution, then repeated the computation for 20 time. Finally the results were averaged.

In Fig. 7 we show the absolute errors for the reconstructed μ_2 represented as black bars for each grid point. The left picture illustrates the true absolute error defined as $|\mu_{2,m} - \mu_2(x_m)|$. The right one depicts the estimated uncertainty according to Björk (22). We see that the calculated μ_2 is in agreement with the assigned (26) within an accuracy of 10^{-10}. Notice that the Björk error estimates does not use the given $\mu_2(x)$ and have almost the same order of magnitude of the true, albeit they represent lower bounds. For μ_1 and a we get similar results not shown here.

This result shows that the computation magnifies the input data uncertainty with a factor 10^4. This information is useful for the estimation of the accuracy of f^n_m or the size of a MC bunch in order to predict the accuracy of the reconstructing parameters, e.g. as follows. From (11) and supposing $\bar{f} \propto 1/N_x^2$, we get the relative accuracy $\varepsilon_r = \varepsilon/\bar{f} \approx N_x/\sqrt{3\alpha R}$, so that

$$R = \frac{1}{3\alpha}\left(\frac{N_x}{\varepsilon_r}\right)^2.$$

If we wish $\varepsilon_r(\mu) \approx 10^{-2}$, due to the amplification factor 10^4 we need $\varepsilon_r(\bar{f}) \approx 10^{-6}$ and, assuming $N_x = 14$ and $\alpha = 0.2$, we estimate $R \propto 10^{16}$. Anyway, this value is excessive because the Tchebicev bound is not very sharp.

Notice that the magnify factor depends on the shape of the distribution data $f^n_{m,l}$, but the analysis of this relationship is beyond the scope of this paper.

Furthermore, in Table 2 in the first line we report the average square error of Fig. 7 of the calculated μ_2, for the true error (2nd column) and Björk estimate (3rd column). In the second line we report the estimated error for the second test. We see a similar drop of accuracy of magnitude 10^4 between the input data and output values μ_2 of the algorithm.

V. Conclusions

A technique for the inference of the parameters of RWs using the Chapman-Kolmogorov equation with transient probability distribution was presented. Moreover, *a posteriori* punctual error estimates of the reconstructed values was calculated. For the 1D case two algorithms were formulated, the first calculates the solution along the states of the RW and have better accuracy than the second, that compute locally on the state by using an interval of sampled time values. The latter algorithm is extended to the 2D case. Numerical tests show the ability of the proposed techniques to deal with the reconstruction of RW, and to provide the related uncertainty and error amplification estimates.

VI. Acknowledgments

The author thanks Prof. Alfio Borzì (University of Würzburg) for the stimulating discussion on this subject.

References

[1] T. W. Anderson and L. A. Goodman. Statistical inference about Markov chains. *Annals of Mathematical Statistics*, 28(1):89–110, 1957.

[2] M. Annunziato and A. Borzì. Using the Chapman-Kolmogorov equation of random walks to identify drift and diffusion of the Fokker-Planck equation. *International Journal of Scientific and Statistical Computing* (IJSSC), 9(1):1–13, 2025.

[3] D. Aurouret and V. Patileam. Discrete-time Markov chains with random observation, 2025. Preprint, arXiv:2505.12971 [stat.ME].

[4] M. S. Bartlett. The frequency goodness of fit test for probability chains. *Proceedings of the Cambridge Philosophical Society*, 47:194–206, 1951.

[5] G. Bernstein and D. Sheldon. Consistently estimating Markov chains with noisy aggregate data. *In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics* (AISTATS), volume 41 of *Proceedings of Machine Learning Research*, pages 1095–1103. PMLR, 2016.

[6] P. Billingsley. Statistical methods in Markov chains. *The Annals of Mathematical Statistics*, 32(1):12–40, 1961.

[7] Å. Björck. Component-wise perturbation analysis and error bounds for linear least squares solutions. *BIT Numerical Mathematics*, 31:238–244, 1991.

[8] T. Breitenbach, M. Annunziato, and A. Borzì. On the optimal control of a random walk with jumps and barriers. *Methodology and Computing in Applied Probability*, 20(1):435–462, 2018.

[9] J. Chodera, N. Singhal, V. Pande, K. Dill, and W. Swope. Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. *Journal of Chemical Physics*, 126:155101, 2007.

[10] D. R. Cox and H. D. Miller. *The Theory of Stochastic Processes*. Chapman & Hall, 1992.

[11] W. Cygan and S. Šebek. Transition probability estimates for subordinate random walks. *Mathematische Nachrichten*, 294:518–558, 2021.

[12] A. DasGupta. Best constants in Chebyshev inequalities with various applications. *Metrika*, 51:185–200, 2000.

[13] P. Diaconis and S. W. W. Rolles. Bayesian analysis for reversible Markov chains. *The Annals of Statistics*, 34(3):1270–1292, 2006.

[14] A. Ghosh. Robust parametric inference for finite Markov chains. *TEST*, 31:118–147, 2022.

[15] J. P. Kaipio and E. Somersalo. Statistical and Computational Inverse Problems. *Springer Series: Applied Mathematical Sciences*. Springer, New York, 2005.

[16] J. D. Kalbfleisch, J. F. Lawless, and W. M. Vollmer. Estimation in Markov models from aggregate data. *Biometrics*, 39:907–919, 1983.

[17] L. Leskelä and M. Dreveton. Robust estimation of a Markov chain transition matrix from multiple sample paths, 2025. Preprint, arXiv:2506.20325 [math.ST].

[18] G. Perez-Hernandez, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé. Identification of slow molecular order parameters for Markov model construction. *Journal of Chemical Physics*, 139:015102, 2013.

[19] C. R. Schwantes and V. S. Pande. Improvements in Markov state model construction reveal many non-native interactions in the folding of ntl9. *Journal of Chemical Theory and Computation*, 9:2000–2009, 2013.

[20] The MathWorks, Inc. Matlab (version r2021a) [computer software], 2021. Computer software.

[21] B. Trendelkamp-Schroer and F. Noé. Efficient bayesian estimation of Markov model transition matrices with given stationary distribution. *The Journal of Chemical Physics*, 138:164113, 2013.

[22] B. Trendelkamp-Schroer, H. Wu, F. Paul, and F. Noé. Estimation and uncertainty of reversible Markov models. *The Journal of Chemical Physics*, 143:174101, 2015.

[23] J. V. Uspensky. *Introduction to Mathematical Probability*. McGraw-Hill Book Company, 1937.