



Rare events analysis and computation for stochastic evolution of bacterial populations

Yingxue Su^a, Brett Geiger^b, Ilya Timofeyev^a, Andreas Mang^a, and Robert Azencott^a

^aDepartment of Mathematics, University of Houston, Houston, Texas, USA; ^bDepartment of Mathematics, High Point University, High Point, North Carolina, USA

ABSTRACT

In this article, we develop a computational approach for estimating the most likely trajectories describing rare events that correspond to the emergence of non-dominant genotypes. This work is based on the large deviations approach for discrete Markov chains describing the genetic evolution of large bacterial populations. We demonstrate that a gradient descent algorithm developed in this article results in the fast and accurate computation of most likely trajectories for a large number of bacterial genotypes. We supplement our analysis with extensive numerical simulations demonstrating the computational advantage of the designed gradient descent algorithm over other, more simplified, approaches.

ARTICLE HISTORY

5 September 2023
22 October 2024

KEYWORDS

Discrete-time Markov chain;
bacterial evolution;
large-deviations; gradient
descent

1. Introduction

Many biological systems can be modeled as random complex systems with a large number of individual interacting agents. The genetic evolution of large bacterial or viral populations is one such example. Moreover, in many applications, random mutations play an important role in describing rare events, such as the emergence of non-dominant genotypes with new biological properties. Examples of such rare events have been observed in long genetic experiments of *Escherichia coli*. For instance, variation of resistance levels in bacterial populations without exposure to antibiotics [1], the emergence of genotypes with lower fitness but with a higher degree of adaptation or evolvability [2, 3], or the emergence of wider cell sizes but later reverting to the original cell shape [4] are only a few such examples.

Large-deviations theory provides a unified and efficient framework for studying rare events in a wide range of random models, such as Markov chains (MCs), Gaussian processes, stochastic differential equations, etc. (We refer to [5–13] for examples.) For instance, for random evolution of large populations, interesting large deviation results were obtained in [14–17]. These works study the Darwinian evolution of asexual populations in the vector space of phenotypic traits that are transmitted to offspring with rare variations due to gene mutations. In this model, competition for limited resources forces a roughly periodic selection.

In the present work, we focus on rare random genetic evolutionary events in bacterial populations, such as the emergence of a low-fitness genotype that reaches high frequency in the population. Computing the most likely evolutionary paths realizing such rare genetic events is a complicated mathematical problem, even for simplified stochastic models of bacterial population evolution. The stochastic genetic evolution of large bacterial populations observed in long-term laboratory experiments (see, e.g., [18–23]) has typically been modeled by simplified MC models (see Section 2), which are fitted to experimental data. To analyze rare events for these types of MC models, we developed [24, 25] a large deviations theoretical framework summarized in Section 3.

The numerical computation of the most likely evolutionary paths realizing rare genetic events involves an optimization problem and, thus, is computationally intractable for straight-forward greedy search algorithms as soon as the number of genotypes exceeds four. In this article, we develop and test several algorithms for the numerical computation of the most likely genetic evolution trajectory connecting the initial population histogram H and a terminal histogram G (see also [26]). We demonstrate that the gradient decent algorithm developed in this article (see Section 5) is capable of computing the most likely evolutionary trajectory for a large number (up to 20) of interacting genotypes. In particular, we present a computational example for 10 genotypes and compare computing times with more straight-forward approaches.

1.1. Laboratory experiments on bacterial genetic evolution. For bacteria, such as *Escherichia coli*, genetic evolution has been explored in many long-term laboratory experiments (see, e.g., [2, 3, 19, 27–36]). For instance, in [19, 18], on each day $t \in \mathbb{N}$ a bacterial population with initial large size N cells grows freely during the day until the daily dose of nutrients is exhausted. After nutrient exhaustion, the cell population remains dormant until the end of the day. Therefore, daily growth duration is practically determined by the (fixed) daily dose of nutrients and hence can be considered fixed throughout the experiment. The growth of each sub-population with a particular fixed genotype is determined by the corresponding growth rate, which is related to the fitness of this genotype. In addition, rare random mutations can also occur. At the end of the day t , the population size becomes a large multiple of N , and one selects by dilution a random sub-sample of roughly N cells, which becomes the initial population on the day $t + 1$. Thus, the bacterial population at the beginning of each day is roughly the same size, N . Typical values for N can range from 10^5 to 10^8 . For $g \in \mathbb{N}$ genotypes, one records (daily if feasible) frequencies of cells with each genotype and thus, each population is characterized by its population histogram $H_t = [H_t(1), H_t(2), \dots, H_t(g)] \in \mathbb{R}^g$ of genotype frequencies $H_t(j), j = 1, \dots, g$. For the colony of j -cells (i.e., cells with genotype j), daily growth by cell divisions roughly multiplies its initial size by a fixed growth factor $F_j \gg 1$. We assume that genotypes are ordered by increasing fitness so that $F_1 < F_2 < \dots < F_g$. Thus, the genotype g is called *dominant*, and its frequency $H_t(g)$ tends to 1 for large t . At each j -cell division, the genotype j is typically inherited by the two daughter cells, unless a very rare random genotype mutation from j to $k \neq j$ occurs. Mutations approximately follow a Poisson distribution with a very small mutation rate, typically ranging from 10^{-9} to 10^{-6} .

1.2. Fixation of non-dominant genotypes and rare genetic events. For any non-dominant genotype $j < g$ initially absent in the population, a j -cell colony may emerge for the first time on day t , due to random mutations. However, for large populations of size N , this new

j -cell colony has an extremely small probability of reaching *fixation* at some later day $t + \tau$, i.e., of reaching a high frequency $H_{t+\tau}(j)$ (e.g., higher than 40%). Indeed, this probability vanishes at an exponential rate as N increases. To enable a precise analysis of rare genetic events such as fixation of non-dominant genotypes, we have developed (see [24]) a *large deviation theory* framework underlying the stochastic genetic evolution of large bacterial populations. In our stochastic models, the initial population at the beginning of each day t has the same size N due to random sub-sampling selection at the end of the previous day. Consider any fixed time $T > 0$. The random genetic evolution of the bacterial population over T days is then a random sequence of histograms $\mathbf{H} = \{H_1, \dots, H_T\}$, characterized by a MC over the set $\mathcal{H} \subset \mathbb{R}^g$ of all histograms of dimension g . This MC path space Ω_T is the set of all possible sequences $\mathbf{H} = \{H_1, \dots, H_T\}$ of population histograms. For large $N > 10^5$ and fixed initial histogram H , the probability distribution P_N of the random paths \mathbf{H} starting at $H_1 = H$ is highly concentrated around a single path, namely the unique *mean trajectory* $\mathbf{M} = \{M_t, t = 1, 2, \dots, T\}$ starting at H (i.e., $M_1 = H$).

In a large deviation theory framework, a key step is to determine the *rate functional*

$$\lambda : \Omega_T \rightarrow \mathbb{R}^+,$$

which controls the probabilities of rare genetic events (we define λ in (6)). For each population trajectory $\mathbf{H} \in \Omega_T$ starting at $H_1 = H$, we have computed the limit

$$\lim_{N \rightarrow \infty} \frac{1}{N} \log P_N(\mathbf{H}) = -\lambda(\mathbf{H}) \leq 0.$$

Whenever \mathbf{H} is not identical to the mean trajectory \mathbf{M} starting at $H_1 = H$, one has $\lambda(\mathbf{H}) > 0$ and the probability $P_N(\mathbf{H})$ is a vanishing exponential of the order of $\exp(-N\lambda(\mathbf{H}))$.

1.3. Contributions. Given any two population histograms H and G , the *most likely* random population path $\mathbf{H} \in \Omega_T$ such that $H_1 = H$ and $H_T = G$ is the deterministic path \mathbf{H}^* minimizing the cost function $\lambda(\mathbf{H})$ over all paths $\mathbf{H} \in \Omega_T$ such that $H_1 = H$ and $H_T = G$. *The main focus of this paper is developing and testing practical algorithms enabling computation of the most likely path \mathbf{H}^* connecting two given histograms H and G in T steps.* This task can be quite challenging for straightforward algorithms when the number of genotypes g exceeds four. This problem has several natural analogies with the computation of a geodesic connecting two points on a smooth Riemannian manifold. In particular, we have identified a recursive reverse-time equation for computing the optimal trajectory \mathbf{H}^* given the final and penultimate histograms. Since the penultimate histogram is not known a priori, this leads to an optimization problem with respect to the penultimate histogram. We present several algorithms to numerically compute \mathbf{H}^* for $3 \leq g \leq 20$. We also would like to point out that computing these rare paths is crucial for estimating fixation probabilities by *importance sampling* simulations. For instance, after the most likely path has been computed, one can develop Monte-Carlo importance sampling simulations for trajectories that are close to the most likely path and, thus, efficiently estimate probabilities of rare events (e.g., probability of reaching the final histogram). The code is available at [37].

1.4. Manuscript organization. In the next section, we describe the MC model and discuss the three daily stages for bacterial populations: growth, mutations, and dilution. In [Section 3](#), we discuss the main results from the large deviations theory. In particular, we introduce the one-step cost function (15) and the formula for the reverse computation of the most

likely trajectory (22). We present the two simpler algorithms for the computation of the most likely evolutionary trajectories and discuss their efficiency in Section 4. The gradient descent algorithm is presented in Section 5. Conclusions are presented in Section 6 and Appendix A, B, and 5.3 summarizes the main notation used in this article, computational parameters, and the gradient descent formulas, respectively.

2. Simplified genetic evolution model for bacterial populations

The MC model of population evolution studied here, as well as in [24], has been applied to emulate the stochastic evolution of large bacterial populations observed in long-term laboratory experiments [18]. We assume that there is a fixed number of possible genotypes denoted as $\{1, 2, \dots, g\}$. In cell population $pop(t)$ observed at the beginning of day t , the initial frequency of j -cells (cells with genotype j) is denoted as $H_t(j)$, and the population histogram $H_t = [H_t(1), \dots, H_t(g)]$ characterizes the state of $pop(t)$. On day t , the initial population $pop(t)$ goes through three successive phases: (i) deterministic growth, (ii) random mutations, and (iii) random selection of a subsample of fixed size N , which then becomes $pop(t + 1)$. Hence, all populations $pop(t)$, $t = 1, 2, \dots$, have the same fixed (large) size N .

Definition 2.1. The set of all possible histograms $\mathcal{H} = \{H \in \mathbb{R}^g\}$ are vectors of length g such that $0 \leq H(j) \leq 1$ and $\sum_j H(j) = 1$. Please note that $\mathcal{H} \subset \mathbb{R}^g$ is compact and convex.

2.1. First phase: daily deterministic growth

During the deterministic growth, the number of j -cells increases from initial size $NH_t(j)$ to the final size $NH_t(j)F_j$, where $F_j > 1$ is a fixed multiplicative *growth factor*. The growth factor can be computed as $F_j = \exp(\Delta t \times a_j)$, where Δt is the duration of deterministic daily growth (assumed fixed in our model) and a_j is the fitness of genotype j . We assume that genotypes are ordered by their fitness, i.e., the vector $F = [F_1, \dots, F_g]$ of growth factors is ordered by increasing fitness, so that $F_1 < \dots < F_g$, and g is called the *dominant genotype*. At the end of the growth phase, $pop(t)$ reaches the size $N\langle F, H_t \rangle$, and the initial histogram H_t becomes $\Phi(H_t)$, where the function $\Phi : \mathcal{H} \rightarrow \mathcal{H}$ is given by

$$\Phi_j(H) = F_j H(j) / \langle F, H \rangle \quad \text{for all } j = 1, \dots, g, \text{ and } H \in \mathcal{H}. \quad (1)$$

Here, $\langle \cdot, \cdot \rangle$ denotes the standard inner product of two vectors in \mathbb{R}^g .

2.2. Second phase: daily random mutations

In bacterial evolution, each time a cell splits into two new cells, random mutations between genotypes may occur with very small probabilities. We have studied this type of daily dynamics using stochastic differential equations with Poisson noise in [38]. However, in this work, focused on the computational aspects of rare event analysis, we simplify the daily population dynamics by assuming that all random mutations occur simultaneously at the end of each daily deterministic growth period.

In the model considered here, during the day t mutation phase, all individual cells may randomly mutate, independently of each other. Each j -cell has a very small probability $m \leq 10^{-6}$ of mutating, and the conditional (given that a mutation occurs) probability of a j -cell mutating into a k -cell is given by a fixed number $0 \leq q_{j,k} \leq 1$ for $k \neq j$ and $q_{j,j} = 0$. For

each j , we assume that, given H_t , the random number of mutants emerging among j -cells is Poisson distributed with mean $mNF_jH_t(j)$. Thus, the key mutation parameters are

- a small *mutation rate* $0 < m \leq 10^{-6}$,
- the $g \times g$ matrix $Q = \{q_{j,k}\}$ of genotype transition probabilities with $\sum_k q_{j,k} = 1$ and $q_{j,j} = 0$.

Denote $R \in \mathbb{R}^g \times \mathbb{R}^g$ a random matrix with $R_{j,k}$ describing the number of j -cells mutating into k -cells on day t . The main diagonal of the mutation matrix R is equal to 0, and its conditional mean, given $H_t = H$, verifies

$$\mathbb{E}(R/N \mid H_t = H) = \rho(H) \quad \text{with coefficients} \quad \rho(H)_{j,k} = mq_{j,k}F_jH(j), \quad (2)$$

where R/N denotes a normalized random mutation matrix with N being the number of cells in the population at the beginning of each day.

2.3. Third phase: daily random selection

After deterministic growth and random mutations, the population $pop(t)$ becomes a much larger population $POP(t)$ of size $N\langle F, H_t \rangle$. Typically, in *E. coli* laboratory experiments, $\text{size}(POP(t)) \gg 200N$. At the end of the day t , the *daily selection phase* is implemented by selecting from $POP(t)$ a *random sample* of fixed size N and denoting this sample as $pop(t+1)$. On day $t+1$, $pop(t+1)$, in turn, undergoes growth, mutation, and end-of-day random selection of N cells. For *E. coli* experiments, daily random selections are often implemented by dilution of $POP(t)$. Next, we introduce some notation and discuss a mathematical model for dilution.

Definition 2.2. Matrix A is called *N-rational* if for a positive integer N all the coefficients of NA are non-negative integers.

Given $H_t = H$ and $R/N = r$, the random population histogram J_t of $POP(t)$ is given by

$$J_t(j) = \Psi_j(H, r) = \frac{1}{\langle F, H \rangle} \left(H(j)F_j - \sum_k r_{j,k} + \sum_k r_{k,j} \right). \quad (3)$$

Here, $N\langle F, H \rangle \sum_k r_{j,k}$ and $N\langle F, H \rangle \sum_k r_{k,j}$ is the total number of cells mutating “out” of and “into” the genotype j , respectively. The random histograms H_t and the normalized random mutation matrices $r = R/N$ are *N-rational* by construction.

Since $pop(t+1)$ is a random subsample of size N extracted from $POP(t)$, the conditional distribution of NH_{t+1} given the histogram $J_t = J$ of $POP(t)$ is a multinomial distribution $\mu_{N,J}$ defined by

$$\mu_{N,J}(v) = N! \prod_{j=1 \dots g} \frac{(J(j))^{v(j)}}{v(j)!}$$

for any vector $v = [v(1), \dots, v(g)]$ of non negative integers with $\sum_j v(j) = N$. More precisely, for any *N-rational* histogram G , one has

$$P(H_{t+1} = G \mid J_t = J) = P(NH_{t+1} = NG \mid J_t = J) = \mu_{N,J}(NG).$$

2.4. Markov chain of population histograms: mean trajectories

Definition 2.3. H is an interior histogram if $H \in \mathcal{H}$ and $H(j) > 0$ for all $j = 1, \dots, g$.

Definition 2.4. The distance between two histograms $H, H' \in \mathcal{H}$ is given by

$$\|H - H'\| = \sup_{j=1, \dots, g} |H(j) - H'(j)|.$$

The preceding succession of three daily phases (growth, mutations, and selection) defines a stochastic process H_t as a discrete Markov chain $\Pr(H_{t+1} = H' \mid H_t = H)$ with trajectories taking values in the compact convex set $\mathcal{H} \subset \mathbb{R}^g$ of population histograms, where $t = 1, 2, \dots$, indexes successive days. Using the mathematical formalism for the three daily stages discussed above, one can compute transition probabilities $\Pr(H_{t+1} = H' \mid H_t = H)$ for any histograms $H', H \in \mathcal{H}$. Since at the beginning of each day, we consider populations of fixed size, N , all trajectories consist of N -rational g -dimensional vectors and, thus, the Markov chain H_t has a finite number of states. The path space Ω_T of this Markov chain for $1 \leq t \leq T$ is the set of all sequences $\mathbf{H} = \{H_1, \dots, H_T\}$ of N -rational population histograms, endowed with the distance $\|\mathbf{H} - \mathbf{H}'\| = \max_{t=1, \dots, T} \|H_t - H'_t\|$.

The *mean trajectory* $M_t = \mathbb{E}(H_t \mid H_1 = H)$ is recursively computed as [24]

$$M_{t+1}(H) = f(M_t(H)), \quad (4)$$

where the function f is the *one-step conditional mean* $f(H) = \mathbb{E}(H_{t+1} \mid H_t = H)$ given by

$$f_j(H) = \frac{1}{\langle F, H \rangle} \left(F_j H(j) - m \sum_k q_{j,k} F_j H(j) + m \sum_k q_{k,j} F_k H(k) \right), \quad j = 1 \dots g. \quad (5)$$

For a fixed population size N , the mean trajectory \mathbf{M} verifies $\lim_{t \rightarrow \infty} M_t = \hat{H}$, where the histogram \hat{H} is given by $\hat{H}(g) = 1$ and $\hat{H}(j) = 0$ for all $j < g$. Thus, the limit mean histogram \hat{H} exhibits a fixation of the genotype g having the highest daily growth factor F_g .

The key parameters of this Markov chain model are a large population size $N \geq 10^5$ and a small mutation rate $m \leq 10^{-6}$. The biological context is characterized by the fixed set of evolution parameters $\{g, F, Q\}$, where

- g is the number of genotypes,
- $F = [F_1, \dots, F_g]$ is the vector of ordered daily growth factors, and
- Q is a $g \times g$ matrix with $q_{j,k}$ being the conditional (given that a mutation occurs) probability that a mutant j -cell becomes a k -cell.

For instance, in the *E. coli* experiments [19, 18], parameters of the Markov chain H_t can typically be selected in the following range $10^5 \leq N \leq 10^8$, $10^{-9} \leq m \leq 10^{-6}$, $5 \leq g \leq 10$, and $200 \leq F_1 < \dots < F_g \leq 500$. For many genetic experiments, the conditional probability $q_{j,k}$ that a mutant j -cell becomes a k -cell cannot be directly estimated from data. Hence, for simplicity, we assume uniform genotype transition probabilities $q_{j,k} = 1/(g-1)$ for all $k \neq j$ with $q_{j,j} = 0$.

3. Rare events and large deviation asymptotics

In a previous paper [24], we have rigorously studied the large deviation asymptotics for the Markov chain of population histograms. Those results are applicable as soon as $N \geq 10^5$

and $m \leq 10^{-6}$, and we now recall the main results proved in [24]. Fix any $T > 0$. Any random events of interest concerning trajectories of the Markov chain can be defined as a subset $\mathcal{E} \subset \Omega_T$. For increasing $N \geq 10^5$ and fixed initial histogram $H_1 = H$, the random paths \mathbf{H} starting at $H_1 = H$ have a probability distribution P_N on Ω_T , which becomes highly concentrated around the mean trajectory \mathbf{M} starting at H . Indeed, any closed subset \mathcal{E} of Ω_T such that $\mathbf{M} \notin \mathcal{E}$ has probability $P_N(\mathcal{E})$ vanishing at exponential speed as $N \rightarrow \infty$. For large N , \mathcal{E} becomes a rare event and we introduce a more precise definition.

Definition 3.1. For fixed initial histogram $H_1 = H$ and time $T > 1$ we call $\mathcal{E} \subset \Omega_T$ a rare event if $\frac{1}{N} \log(P_N(\mathcal{E})) \rightarrow -\Lambda(\mathcal{E})$ as $N \rightarrow \infty$. Here, $\Lambda(\mathcal{E}) = \min_{\mathbf{H} \in \mathcal{E}} \lambda(\mathbf{H})$, where $\lambda(\mathbf{H})$ is the rate functional (or the cost of the path \mathbf{H}) defined in (6). As a short-hand notation we then write $P_N(\mathcal{E})$ is log-equivalent to $\exp(-N\Lambda(\mathcal{E}))$.

3.1. Most likely histogram trajectory linking two population histograms

Definition 3.2. A path $\mathbf{H} = \{H_1, \dots, H_T\} \in \Omega_T$ is called an *interior path* if all histograms H_t are interior histograms, i.e., they verify $H_t(j) > 0$ for $j = 1, \dots, g$.

Denote $\mathcal{T}_N \subset \Omega_T$ the *thin tube* of all paths in Ω_T lying within distance $1/N$ of \mathbf{H} . Then (see [24]), as $N \rightarrow \infty$ the probabilities $P_N(\mathcal{T}_N)$ are definition *log-equivalent* to $\exp(-N\lambda(\mathbf{H}))$, where the *rate function* $\lambda(\mathbf{H}) \geq 0$ is given by

$$\lambda(\mathbf{H}) = \sum_{t=1}^{T-1} C(H_t, H_{t+1}). \quad (6)$$

Here, $C(H, G) \geq 0$ is a computable one-step cost function (see (13) below).

The *most likely path* $L(H, G) = \{L_1 = H, L_2, \dots, L_T = G\}$ connecting histograms H and G in T steps is then determined by minimizing $\lambda(\mathbf{H})$ over all paths $\mathbf{H} = [H_1, \dots, H_T]$ such that $H_1 = H$ and $H_T = G$, i.e.,

$$L(H, G) = \mathbf{H}^* = \arg \min_{\mathbf{H} \in \Omega_T^+(H, G)} \lambda(\mathbf{H}),$$

where

$$\Omega_T^+(H, G) = \{\mathbf{H} \in \Omega_T : H_1 = H \text{ and } H_T = G\} \quad (7)$$

is a restricted space of paths.

As shown in [24], any most likely path $\mathbf{H}^* = [H_1^*, \dots, H_T^*]$ must verify a second order recursive equation expressing explicitly H_t^* in terms of H_{t+1}^*, H_{t+2}^* . This article explores efficient strategies for computing the path \mathbf{H}^* since such numerical computations are particularly challenging if the number of genotypes g exceeds 4.

3.2. The one-step cost function $C(H, G)$

On each day t , the initial random histogram H_t , the normalized matrices $r_t = R_t/N$ of random mutations, and the histogram J_t of POP_t after mutations, must verify the deterministic relation $J_t = \Psi(H_t, r_t)$ in (3). Since the number $N\mu_t(j)$ of mutants among the j -cells of POP_t is inferior to the size $NF_j H_t(j)$ of the j -cells colony, r_t must belong to the set $\mathcal{M}(H_t)$ of $g \times g$

matrices with entries $r_{j,k}$ defined by

$$\mathcal{M}(H) = \left\{ r : r_{j,k} \geq 0; \quad r_{jj} = 0; \quad \text{and} \quad \sum_{k=1}^g r_{j,k} \leq H(j)F_j \quad \text{for all } j \right\}. \quad (8)$$

Thus, $\mu_t(j)$ have independent slightly truncated Poisson distributions with means $mF_jH_t(j)$.

Fix any interior histogram H and any matrix $r \in \mathcal{M}(H)$. Denote $J = \Psi(H, r)$. As a short-hand notation, we use $H_t \approx H$, $r_t = R_t/N \approx r$, whenever the sup-norms $\|H_t - H\|$ and $\|r_t - r\|$ are bounded by c/N for some fixed constant $c > 0$. One then has a similar bound for $\|J_t - J\|$, for which we also write $J_t \approx J$. The rate function controlling large deviations from the mean for products of truncated Poisson distributions is determined in [24] to show that for any interior histogram H , any matrix $r \in \mathcal{M}(H)$, and for $N \rightarrow \infty$ one has

$$\Pr(r_t \approx r \mid H_t \approx H) \text{ is log- equivalent to } \exp(-Nu(r, H)), \quad (9)$$

where

$$u(r, H) = \sum_{j,k} r_{j,k} \log r_{j,k} - r_{j,k} [1 + \log(mq_{j,k}F_jH(j))] + mq_{j,k}F_jH(j). \quad (10)$$

Given J_t , the conditional distribution of NH_{t+1} is a multinomial distribution with mean NJ_t , and its large deviations from the mean are controlled by an explicit rate function. Indeed, for any two interior histograms (G, J) , and for $N \rightarrow \infty$, one has [24]

$$\Pr(H_{t+1} \approx G \mid J_t \approx J) \text{ is log- equivalent to } \exp(-NKL(G, J)), \quad (11)$$

where the *Kullback-Leibler divergence* $KL(G, J) \geq 0$ is finite, and given by

$$KL(G, J) = \sum_j G(j) \log \frac{G(j)}{J(j)} \quad (12)$$

Since $J = \Psi(H, r)$, one has

$$\Pr(\{H_{t+1} \approx G\} \text{ and } \{r_t \approx r\} \mid H_t \approx H) = \Pr(H_{t+1} \approx G \mid J_t \approx J)P(r_t \approx r \mid H_t \approx H)$$

so that, due to (11) and (9),

$$\Pr(\{H_{t+1} \approx G\} \text{ and } \{r_t \approx r\} \mid H_t \approx H) \text{ is log-equivalent to } \exp(-Nv(H, r, G)),$$

where $v(H, r, G) = u(H, r) + KL(G, \Psi(H, r))$.

The approximate one-step transition probability $\Pr(\{H_{t+1} \approx G\} \mid H_t \approx H)$ is then log-equivalent to the sum over all $r \in \mathcal{M}(H)$ of the vanishing exponentials $\exp(-Nv(H, r, G))$. This sum is log-equivalent to the largest of all these terms, namely $\exp(-NC(H, G))$, where the *one-step cost function* $C(H, G) \geq 0$ is defined by

$$C(H, G) = \min_{r \in \mathcal{M}(H)} v(H, r, G) = \min_{r \in \mathcal{M}(H)} [u(r, H) + KL(G, \Psi(H, r))]. \quad (13)$$

Therefore, for arbitrary interior histograms H, G , and $N \rightarrow \infty$, we have the asymptotic result

$$\Pr(\{H_{t+1} \approx G\} \mid H_t \approx H) \text{ is log-equivalent to } \exp(-NC(H, G)). \quad (14)$$

The cost $C(H, G) \geq 0$ can only take the value 0 when $G = f(H) = \mathbb{E}(H_{t+1} \mid H_t = H)$, with $f(H)$ given by (5). Hence, for H fixed and large N , the Markov transition kernel is concentrated

at exponential speed (in N) on a fixed, small neighborhood of $f(H)$. When H, G are *interior* histograms, i.e., when $\min(H) > 0$ and $\min(G) > 0$, the cost $C(H, G)$ is a finite smooth function of H and G . Moreover, for a small enough mutation rate $m \leq 10^{-6}$, the cost $C(H, G)$ has an explicit expansion in m given by

$$C(H, G) = KL(G, \Phi(H)) + m \sum_{j,k} F_j H(j) q_{j,k} [1 - U_k/U_j] + O(m^2), \quad (15)$$

where $U_j = \exp(G_j/(F_j H(j)))$ and $\Phi \in \mathbb{R}^g$ with $\Phi(j) = F_j H(j)/\langle F, H \rangle$.

3.3. Large deviation asymptotics in path space

The path space Ω_T of the Markov chain H_t is endowed with a natural distance (see Section 2.4). For any path $\mathbf{H} = [H_1, \dots, H_T]$ in Ω_T and any small $\epsilon > 0$ define the closed “tube” of paths $\mathcal{T}(\mathbf{H}, \epsilon)$ centered at the path \mathbf{H} by

$$\mathcal{T} = \{\mathbf{H}' \in \Omega_T \text{ such that } \|\mathbf{H}' - \mathbf{H}\| \leq \epsilon\}. \quad (16)$$

We say that \mathcal{T} is an *interior tube* if all $\mathbf{H}' \in \mathcal{T}$ are interior paths. We extend the one-step cost $C(H, G)$ to a *multi-steps* cost functional $\lambda(\mathbf{H}) \geq 0$ defined for any $\mathbf{H} \in \Omega_T$ by

$$\lambda(\mathbf{H}) = \sum_{t=1}^{T-1} C(H_t, H_{t+1}). \quad (17)$$

The rate function $\lambda(\mathbf{H})$ defined for paths $\mathbf{H} \in \Omega_T$ can be formally extended to a rate functional $\Lambda(A) \geq 0$ defined for all subsets $A \subset \Omega_T$ by

$$\Lambda(A) = \inf_{\mathbf{H} \in A} \lambda(\mathbf{H}). \quad (18)$$

The cost function $C(H, G)$ is continuous in H and G on the set of interior histograms. This implies that when \mathbf{H} is any fixed interior path, then $\Lambda(\mathcal{T}(\mathbf{H}, \epsilon)) \rightarrow \lambda(\mathbf{H})$ as $\epsilon \rightarrow 0$. As $N \rightarrow \infty$, we proved in [24] that $\lambda(\mathbf{H})$ is the large deviations *rate functional* controlling probabilities of rare events for our Markov chain. This result is formulated as follows.

Theorem 3.1. *Fix an interior path $\mathbf{H} = [H_1, \dots, H_T] \in \Omega_T$, and an interior tube of paths $\mathcal{T}(\mathbf{H}, \epsilon)$. Denote $\mathbf{h} = [h_1, \dots, h_T]$ a random path of population histograms starting at $h_1 = H_1$. Then, as $N \rightarrow \infty$, the probabilities $P_N(\mathbf{h} \in \mathcal{T})$ are log-equivalent to $\exp(-N\Lambda(\mathcal{T}))$, where the set functional Λ is defined by (18). More precisely, the “thin” tubes of paths $\mathcal{T}(\mathbf{H}, 1/N)$ have probabilities $P_N(\mathbf{h} \in \mathcal{T})$ which are log-equivalent to $\exp(-N\lambda(\mathbf{H}))$ as $N \rightarrow \infty$.*

Proof See [24]. □

The non-negative rate functional $\lambda(\mathbf{H})$ can only reach the value 0 when \mathbf{H} is the mean trajectory \mathbf{M} starting at H_1 (see (4)). Thus, mean trajectories are also called *zero-cost paths*. When the initial condition of the Markov chain is fixed at H_1 and as $N \rightarrow \infty$, the probabilities $P_N(\mathbf{H})$ vanish exponentially fast if \mathbf{H} is not the zero-cost path \mathbf{M} starting at H_1 but tends to 1 exponentially fast if $\mathbf{H} = \mathbf{M}$.

Consider fixed histograms H and G and recall the definition of the space of restricted paths $\Omega_T^+(H, G)$ in (7). Consider paths $\mathbf{H} \in \Omega_T^+(H, G)$ and the corresponding set of all tubes

$\mathcal{T}(H, 1/N)$. Then, the most likely tube $\mathcal{T}(H^*, 1/N)$ is centered around the path H^* which solves the minimization problem

$$H^* = \arg \min_{H \in \Omega_T^+(H, G)} \lambda(H), \quad (19)$$

where $\lambda(H)$ is given by (17).

For interior histograms H and G , this minimization problem always has at least one solution H^* which depends on the number of steps $T \geq 1$. Any such minimizing path is a *most likely path* linking H to G in T steps. Indeed, for the Markov chain of population histograms, and for large N , following the thin tube $\mathcal{T}(H^*, 1/N)$ in path space maximizes the probability of reaching histogram G at time T given $H_1 = H$. In fact, for large N , both $\Pr(H_T = G \mid H_1 = H)$ and $\Pr(H \in \mathcal{T}(H^*, 1/N))$ are *log-equivalent* to $\exp(-N\lambda(H^*))$. Note that the optimal rate $\lambda(H^*)$ also depends on T .

In long-term studies of bacterial evolution, one may know that the population evolved from a remote past ancestor histogram H to a currently observed histogram G , without knowing the precise time duration between these two observations. This leads to the problem of finding both the unknown time T and the *most likely path* linking H to G in T steps. In our numerical benchmark studies below, we implemented this type of computation by numerically minimizing in T the optimal rate $\lambda(H^*)$. Our computational strategies outlined in this article yield both an optimal number of steps T^* and a most likely path $H^*(T^*) \in \Omega_{T^*}^+(H, G)$. The uniqueness of T^* and $H^*(T^*)$ has not been proved but our numerical computations of $H^*(T^*)$ for many random pairs (H, G) indicate that uniqueness is likely to hold for almost all interior histograms H and G .

3.3.1. Analogy between most likely population paths and riemannian geodesics. For many classes of continuous time Markov processes in \mathbb{R}^k , large deviation rate functions analogous to λ have been explicitly determined. For instance, for stochastic differential equations (SDEs) with noise multiplied by a small parameter ϵ , denote $p_\epsilon(f)$ the probability that a random SDE path remains within a thin tube around the smooth path $f_t \in \mathbb{R}^k$ for all $t \leq T$. Then (see [10, 7]), $p_\epsilon(f)$ is roughly equivalent to $\exp(-\lambda(f)/\epsilon^2)$, with a rate function $\lambda(f) \geq 0$ given by

$$\lambda(f) = \int_{[0, T]} \text{Cost}(f_t, f'_t) dt \quad \text{when } f'_t \in L_2(0, T), \quad (20)$$

where $\text{Cost}(u, v) \geq 0$ is an explicit smooth function of $u, v \in \mathbb{R}^k$. Given $x, y \in \mathbb{R}^k$, the most likely SDE path f such that $f_0 = x$ and $f_T = y$, is determined by minimizing $\lambda(f)$ over all paths f subject to $f_0 = x$ and $f_T = y$. All minimizing paths f^* must verify $\partial_f \lambda(f) = 0$, where $\partial_f \lambda$ is the differential of $\lambda(f)$. This yields a non-linear, second-order differential equation verified by f^* for all t , to be solved under the constraints $f_0^* = x, f_T^* = y$.

On any Riemannian manifold S , all *geodesics* $f_t \in S$ must verify an analogous second-order ordinary differential equation (ODE). Indeed, a *geodesic* connecting $f_0 = x$ to $f_T = y \neq x$ in S is a minimizing path for the *kinetic energy* $Kin(f)$ defined by

$$Kin(f) = \frac{1}{2} \int_{[0, T]} \langle f'_t, \gamma_{f_t} f'_t \rangle dt \quad \text{when } f'_t \in L_2(0, T), \quad (21)$$

where the quadratic form γ_z is the Riemann metric at $z \in \mathcal{S}$. Note that if one sets

$$\text{Cost}(f_t, f'_t) = \frac{1}{2} \langle f'_t, \gamma_{f_t} f'_t \rangle$$

then $\text{Kin}(f)$ has the same form as $\lambda(f)$ given in (20).

Minimizing the kinetic energy $\text{Kin}(f)$ as well as minimizing the SDE rate function $\lambda(f)$ over all smooth paths f_t such that $f_0 = x$ and $f_T = y$ lead to solving two similar second-order ODEs with endpoint constraints. Similarly, minimizing our rate functional $\lambda(\mathbf{H})$ over all discrete-time paths $\mathbf{H} \in \Omega_T^+(H, G)$ leads (see the next section) to solving in reverse time a recursive equation of order two, analogous to the second-order ODEs for continuous-time problems discussed above. Thus, in this article, the most likely path $\mathbf{H}^* \in \Omega_T^+(H, G)$ connecting two population histograms H and G will also be called a *geodesic* from H to G .

3.4. Reverse time computation of most likely paths

Any most likely path $\mathbf{H}^* \in \Omega_T^+(H, G)$ must solve the cost minimization problem (19). As proved in [24] for small mutation rate m , any such minimizing path $\mathbf{H}^* \in \Omega_T^+(H, G)$ is *fully determined* by its last two histograms $H_T^* = G$ and H_{T-1}^* . Indeed, \mathbf{H}^* must verify the *reverse time recursive equation*

$$H_t^* = \chi(H_{t+1}^*, H_{t+2}^*) \quad \text{for all } 1 \leq t \leq T-2, \quad (22)$$

where $\chi(y, z)$ is an explicit histogram-valued smooth function defined for all interior histograms (y, z) . For small mutation rate $m \leq 10^{-6}$, the first-order Taylor expansion of $\chi(y, z)$ was computed in [24] as follows:

$$\chi(y, z) \simeq x + O(m^2) \quad \text{with} \quad x_j = Y_j(1 + mw_j) \quad \text{for } j = 1, \dots, g. \quad (23)$$

Here, given the histograms y and z , one computes the interior histogram Y and the vector $w \in \mathbb{R}^g$ by the following explicit formulas (where $j, k = 1 \dots g$)

$$\begin{aligned} X_j &= \frac{y_j}{F_j} \exp\left(\frac{F_j}{\langle F, y \rangle} - \frac{z_j}{y_j}\right) > 0, & Y_j &= \frac{X_j}{\sum_k X_k}, \\ e_{j,k} &= \exp\left(-y_j/(F_j Y_j) + y_k/(F_k Y_k)\right), \\ \alpha_j &= \sum_k \left(q_{j,k} e_{j,k} - \frac{F_k X_k}{F_j X_j} q_{k,j} e_{k,j} \right), \\ f_{j,k} &= \exp\left(-z_j/(F_j y_j) + z_k/(F_k y_k)\right), \\ \beta_j &= F_j \sum_k q_{j,k} - \left(F_j + \frac{z_j}{y_j} \right) \sum_k f_{j,k} q_{j,k} - \frac{z_j}{F_j y_j^2} \sum_k F_k y_k q_{k,j} f_{k,j}, \\ w &= \alpha + \beta - \langle Y, \alpha + \beta \rangle. \end{aligned}$$

The reverse recurrence (22) reduces the computation of \mathbf{H}^* to the optimization search with respect to the penultimate histogram

$$Z^* = \arg \min_{Z \in \mathcal{H}^+} \lambda([H_1 \equiv H, H_2, \dots, H_{T-1} = Z, H_T \equiv G]),$$

where $\mathcal{H}^+ = \{Z : Z \in \mathcal{H} \text{ and } Z(j) > 0\}$ is the set of all possible interior histograms. Then, the optimal trajectory is given by $\mathbf{H}^* = [H, H_2, \dots, H_{T-1} \equiv Z^*, H_T \equiv G]$, where H_t are computed from (22) for $t = T-2, T-3, \dots, 3, 2$.

Note that condition $H_1 = H$ is always enforced; consequently, we use the reverse recursive relationship (22) only until $t = 2$.

We always seek $H_{T-1} = Z^*$ among interior histograms so that the path \mathbf{H} generated by Equation (22) is an interior path. Next, we consider this numerical problem, which is similar to reverse geodesic shooting on Riemannian manifolds, and is quite challenging when $g \geq 4$.

We call the T -steps cost minimizing path $\mathbf{H}^* \in \Omega_T^+(H, G)$ the *geodesic of length T* from H to G , and we denote its (minimized) cost $\lambda(\mathbf{H}^*)$ by $\lambda_T^*(H, G)$. To fully determine the most likely path when the number T of path steps is allowed to be arbitrary, one needs to compute

$$\lambda^*(H, G) = \min_{T \geq 1} \lambda_T^*(H, G)$$

and the optimal number of steps T^* such that $\lambda^*(H, G) = \lambda_{T^*}^*(H, G)$. For given interior histograms H, G , and mutation rates m small enough, we conjecture that a finite T^* always exists, as supported by numerical simulations outlined below.

For any interior geodesic $\mathbf{H} = [H_1, \dots, H_T = G]$, all the intermediary histograms H_t are fully determined by both G and the *penultimate histogram* $Z = H_{T-1}$ due to the reverse recurrence (22). In particular, the initial histogram H_1 is then a smooth function of G and Z defined iteratively by

$$\theta_1(Z, G) = G, \quad \theta_2(Z, G) = Z, \quad \text{and} \quad \theta_t = \chi(\theta_{t-1}, \theta_{t-2}) \quad \text{for } t \geq 3. \quad (24)$$

Here, $\theta_T = [\theta_T, \theta_{T-1}, \dots, \theta_2, \theta_1]$ represents the trajectory $\mathbf{H} = [H_1, \dots, H_T = G]$ in reverse time, i.e., $\theta_T = H_1, \theta_1 = G, \theta_2 = H_{T-1}$, etc. Notice that it is very unlikely that the reverse computation of the most likely path using (24) would amount to $\theta_T = H_1$. Therefore, when searching for the most likely path $\mathbf{H}^* \in \Omega_T^+(H, G)$ we use the reverse formula (24) to compute θ_t for $3 \leq t \leq T$ and then “connect” path θ_T to H using the zero-cost mean path starting at H . Then, the resulting most likely path \mathbf{H}^* is given as a concatenation of two paths—the zero-cost mean path and the path θ_T . Then, \mathbf{H}^* connects H and G by construction and the cost of \mathbf{H}^* is the cost of θ_T plus the cost of the “connection” between the mean path and θ_T . This “connection” can occur at an arbitrary time $2 \leq \tau \leq T$. The key question for the reverse time geodesic shooting is to find a penultimate histogram $Z = H_{T-1}$ and an integer $\tau \geq 2$ such that θ_τ is close to the mean path starting at H .

4. Algorithms for reverse time geodesic shooting

As discussed in the previous section, to compute numerically the most likely path one needs to perform an optimization search for the penultimate histogram $H_{T-1} = Z$. This optimization problem can be combined with the search for the optimal number of steps, T^* . First, we consider the problem of computing the optimal cost $\lambda_T^*(H, G)$ for a fixed number of steps, T .

To this end, we need to fix a set $\mathcal{S} \subset \mathcal{H}^+$ of interior penultimate histograms. (We discuss how to select \mathcal{S} efficiently in Sections 4.2 and 4.3, respectively.) We compute the optimal penultimate histogram by performing an optimization over $Z \in \mathcal{S}$, i.e.,

$$Z^* = \arg \min_{Z \in \mathcal{S}} \lambda([H_1 \equiv H, H_2, \dots, H_{T-1} = Z, H_T \equiv G]),$$

and an approximate geodesic is given by $\mathbf{H}^* = [H_1 \equiv H, H_2, \dots, H_{T-1} = Z^*, H_T \equiv G]$, where H_t are computed from (22) for $t = 2, \dots, T-2$. However, we need to account for the fact that the histogram $H_k = \chi(H_{k+1}, H_{k+2})$ (for $1 < k < T-1$) can be a

boundary histogram. In this case, we cannot continue reverse shooting by using (22). Instead, we connect H_k with a mean trajectory starting at H . During the reverse shooting computation, formula (22) can result in H_k being a boundary histogram. In this case, we have to connect H_{k-1} with the mean trajectory and continue the computation. This can occur for more than one value of k and we connect with the mean trajectory every time H_k ends up on the boundary. This approach is discussed in the next section.

4.1. Computation of approximate geodesics

To define a boundary histogram, we first define a small number $\epsilon(N)$. For practical values $m \leq 10^{-6}$ and $N \geq 10^5$, it can be defined as $\epsilon(N) = 50/N$, so that $\epsilon(N) \leq 5 \times 10^{-5}$. Next, we define a boundary histogram where at least one of the genotypes is nearly extinct.

Definition 4.1. A histogram $H \in \mathcal{H}$ is a *boundary histogram* if $\min_{j=1,\dots,g} H(j) \leq \epsilon(N)$.

Next, to combine the optimization of the cost function for the penultimate histogram $Z \in \mathcal{S}$ and the number of steps, T , we define the maximum possible number of steps in the optimal trajectory, T_{\max} . In practice, one can safely take $T_{\max} \approx 4g, \dots, 5g$

since rare event geodesics with strictly positive cost have a relatively small number of steps (see Section 4.4). The value for T_{\max} above is an approximate practical guideline; longer trajectories can be explored if it is computationally feasible.

Then, for each $Z \in \mathcal{S}$ we perform the reverse shooting using the following two steps:

Step 1. Compute recursively the *open ended* reverse time trajectory θ by the reverse time recursion (22)

$$\theta_1 = G, \quad \theta_2 = Z, \quad \text{and} \quad \theta_t = \chi(\theta_{t-1}, \theta_{t-2}) \text{ for } t \geq 3.$$

The iterative computation of θ_t is stopped at time τ if either $\tau = T_{\max} - 1$ or if θ_τ is a boundary histogram.

Step 2a. If $\tau = T_{\max} - 1$, then θ_τ is an interior histogram and we “connect” the reverse path to H by assigning $\theta_{T_{\max}} = H$.

Step 2b. If $\tau < T_{\max} - 1$, then θ_τ is a boundary histogram and we “connect” the reverse path using the mean trajectory starting at H . In particular, the mean trajectory $\mathbf{M} = [H, M_2, \dots, M_{T_{\max}-\tau}]$ of length $T_{\max} - \tau$ is computed using (4). Notice that the cost of the mean trajectory is always zero. Next, we perform a search for two indexes (u, v) such that cost $\lambda(M_u, \theta_v)$ is minimal among all (u, v) such that $1 \leq u \leq T_{\max} - \tau$ and $1 \leq v \leq \tau$. Notice that this search is fast since T_{\max} is not very large. Then, the trajectory connecting H and G is given by $\mathbf{H} = [H, M_2, \dots, M_u, \theta_v, \dots, \theta_2, G]$ with the cost given by (17). By construction, the length of this trajectory is less or equal to T_{\max} and the penultimate histogram is $\theta_2 = Z$.

Since \mathcal{S} is finite, optimization search over all $Z \in \mathcal{S}$ yields an optimal penultimate histogram $Z^* \in \mathcal{S}$ and the optimal trajectory $\mathbf{H}^* = [H, H_2, \dots, Z^* \equiv H_{T^*-1}, G \equiv H_{T^*}]$ with $T^* \leq T_{\max}$ by construction. Of course, the optimal trajectory \mathbf{H}^* and the number of steps T^* can potentially depend on the choice of the set \mathcal{S} . Thus, it might be necessary to consider

Table 1. Cardinality $\text{card}(\mathcal{S})$ for different choices of the number of genotypes, g , and the discretization level, ζ .

(g, ζ)	(3, 500)	(4, 200)	(5, 100)	(6, 100)	(7, 100)	(8, 100)
$\text{card}(\mathcal{S})$	1.3×10^5	1.4×10^6	$\simeq 10^8$	$\simeq 10^{10}$	$\simeq 10^{12}$	$\simeq 10^{14}$

very large sets of penultimate histograms to find a global minimizing trajectory connecting H and G . Therefore, we discuss heuristic arguments for constructing \mathcal{S} to reduce the number of penultimate histograms and the corresponding computational cost.

4.2. Brute-force search

One obvious choice would be to take $\mathcal{S} = \mathcal{H}^+$, the set of all interior histograms. However, this choice is computationally intractable, and also not necessary. Instead, one can attempt to use a much coarser discretization of \mathcal{H}^+ . Thus, we introduce a parameter ζ with $100 \leq \zeta \leq 500$ and denote \mathcal{H}_ζ^+ the set of all interior ζ -rational histograms. Set \mathcal{H}_ζ^+ provides discretization of \mathcal{H}^+ with mesh size $1/\zeta$ and we can set $\mathcal{S} = \mathcal{H}_\zeta^+$. Note that in this case $\text{card}(\mathcal{S}) \simeq \zeta^{g-1}$. The number of penultimate histograms in \mathcal{S} for different values of g and ζ are listed in Table 1. One can see that it is relatively easy to implement a straightforward search over all $Z \in \mathcal{S}$ for $g \leq 4$ genotypes. However, even for a coarse discretization with $\zeta = 100$, the direct approach becomes computationally intractable (even with a parallelization on 20 nodes) for $g > 5$.

One can attempt to develop an iterative refinement strategy starting with a coarse discretization of, e.g., $\zeta_1 = 20$. Optimization over the set of all penultimate histograms in $\mathcal{S}_1 = \mathcal{H}_{\zeta_1}^+$ will result in an optimal path and optimal penultimate histogram Z_1^* . However, this path is unlikely to be the global minimizer due to a coarse discretization of \mathcal{H}^+ . Then, it is possible to select \mathcal{S}_2 as a small neighborhood around Z_1^* with a finer discretization $\zeta_2 > \zeta_1$. This will result in an optimal penultimate histogram Z_2^* , which can be refined further. This yields an iterative approach for computing the most likely path and the optimal penultimate histogram Z^* . However, this iterative approach is still exceedingly expensive for $g \geq 6$. Therefore, we discuss a different approach for selecting the set \mathcal{S} next.

4.3. p -Quantile approach for constructing \mathcal{S}

In the case of three genotypes, $g = 3$, an extensive computation of geodesics is very fast for $\mathcal{S} = \mathcal{H}_\zeta^+$ with $\zeta = 500$. Thus, for $g = 3$ we performed extensive numerical investigation and computed many most likely paths connecting pairs H and G selected randomly from the set \mathcal{H}^+ . We conjecture that for $g = 3$, most likely paths $\mathbf{H}^* \in \Omega^+(H, G)$ have penultimate histograms Z^* with a fairly low last step cost $C(Z^*, G)$. This suggests that the set \mathcal{S} of penultimate histograms can be selected to minimize the last step cost $C(Z, G)$.

This can be implemented by explicitly computing the derivative of the last step cost and selecting a set \mathcal{S} with a fairly low Euclidian norm $\|\partial_Z C(Z, G)\|$ (see Section 5.3).

Instead of utilizing the derivative of the last step cost, we have developed a more efficient p -quantile approach for selecting \mathcal{S} . This approach is based on the same conjecture of low last-step costs for optimal geodesics. Consider the set \mathcal{H}_ζ^+ of interior histograms that discretizes the set of all interior histograms, \mathcal{H}^+ with step $1/\zeta$. Next, compute one-step costs $C(Z, G)$ for all $Z \in \mathcal{H}_\zeta^+$ and denote β_p the p -quantile of such one-step costs. Then, the set $\mathcal{S} \equiv \mathcal{S}(p)$ of

Table 2. Mean most likely path length computed for $g = 3, 4, 5$.

g	#Geodesics	Mean length
3	10,000	5.02 ± 0.02
4	5,000	6.82 ± 0.04
5	240	4.92 ± 0.05

penultimate histograms can be chosen as

$$S(p) = \{Z \in \mathcal{H}_\zeta^+ \mid C(Z, G) < \beta_p\}. \quad (25)$$

For $\zeta = 100$, the cardinality of $S(p)$ is roughly $p \times 10^{2g-2}$. The choice $S = S(p)$ always reduces the computing time for the brute-force approach discussed in the previous section by a factor of approximately p . Numerical results are presented in Section 4.4.2.

4.4. Numerical results

In this section, we describe numerical results designed to compare the two approaches for constructing the set of penultimate histograms discussed in Sections 4.2 and 4.3, respectively. The main goal is to demonstrate the computational efficiency of the p -quantile approach. We also discuss the biological consequences of the computed most likely paths. The parameters of our model are estimated from laboratory experiments on the genetic evolution of *E. coli* bacteria [28, 39] and are presented in Section B.

All computations in this article were carried out on the Opuntia multi-node cluster available at the University of Houston. Each of these nodes is equipped with 20 core CPUs (Intel Xeon E5-2680v2 2.8 GHz). Each computational task was divided into 20 approximately equal parts, and all computational parts were carried out in parallel. On each node, computational performance is increased by exploiting (hyper-)threading via MATLAB's parallel computing toolbox.

4.4.1. Results for $g = 3, 4, 5$ genotypes

Runtimes for simulations with both approaches for constructing the set S are reasonable for $g = 3, 4, 5$ genotypes and $\zeta = 100$. Thus, we first perform simulations in this parameter regime to compare the approaches presented in Sections 4.2 and 4.3, respectively. We selected 10,000, 5,000, and 240 random pairs (H, G) of terminal histograms for $g = 3, 4$, and 5, respectively. For each pair (H, G) we then used $\zeta = 100$ and performed the search for the optimal trajectory with $S = \mathcal{H}_\zeta^+$.

The distribution of lengths for computed geodesics is described by the statistics reported in Table 2. We also display the histogram of geodesics lengths for $g = 3$ and $g = 4$ genotypes in Figure 1. The average length of geodesics for $g = 3, 4, 5$ genotypes is quite short (at the order of $T = 6$ days). In the context of *E. coli* laboratory experiments, the doubling time for the colony is approximately 20 min under optimal growth conditions. However, optimal conditions cannot be maintained for the whole 24-h period; the actual growth duration is about 9 to 10 h per day (see, e.g., [18]), after that, the population is dormant until the beginning of the next day when a random subsample of fixed size N is generated by dilution, and a fresh daily dose of nutrients is injected. Thus, in practical experimental setups growth factors are in the range [200, 240]. Therefore, most daily cell lineages involve approximately

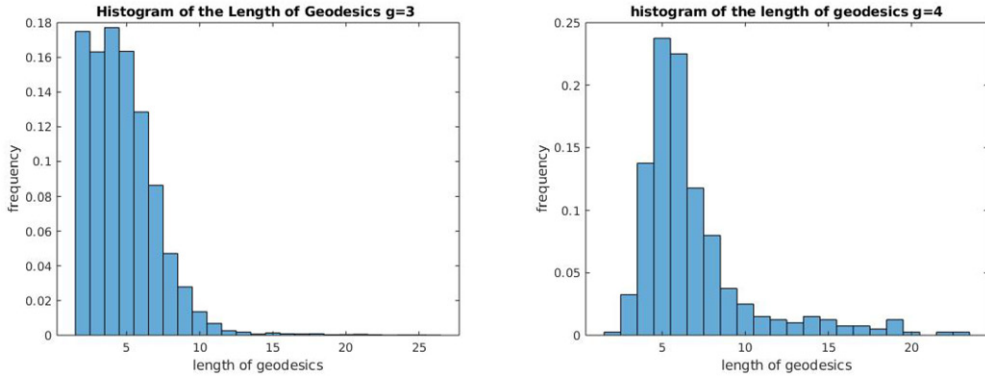


Figure 1. Histogram of most likely paths lengths in computations for $g = 3$ (left) and $g = 4$ (right) genotypes.

Table 3. Efficiency of the p -quantile algorithm for $g = 3, 4, 5$ genotypes.

p	100%	75%	50%	25%	10%	5%	1%
$g = 3$	1	1	1	0.95	0.54	0.35	0.20
$g = 4$	1	1	1	1	0.94	0.66	0.25
$g = 5$	1	1	1	1	1	1	0.71

Efficiency is defined as the fraction of most likely paths coinciding for the p -quantile and straightforward algorithms.

70 successive generations of cells and the most likely path of length T corresponds to about $70 \times T$ generations of bacterial cells.

When the terminal histogram G is fairly different from initial histogram H , but at the same time has a low concentration of the fittest genotype $k = g$, and most likely path connecting H and G is a rare event occurring with a very small probability. A relatively small most likely path length indicates that whenever a rare event of this type is realized, it essentially happens in a succession of a few highly unlikely steps. An analogous phenomenon has been noted for many diffusions with a small noise (e.g., [7]), where the unlikely escape from a stable equilibrium point essentially only occurs when the process trajectory picks at each time point the most unlikely direction, namely the direction opposite to the drift.

Next, we consider the p -quantile approach for generating the set of penultimate histograms, \mathcal{S} . We consider percentiles $p \in \{1\%, 5\%, 10\%, 15\%, 20\%, 25\%, 100\%\}$, and for each pair (H, G) in the random sample we compute the best approximate most likely path using the p -quantile algorithm. We then define the efficiency of the p -quantile algorithm as the fractions of most likely paths that are identical to the straightforward approach. The efficiency of the p -quantile algorithm is reported in Table 3.

Table 3 demonstrates that the efficiency of the p -quantile algorithm increases very quickly as g increases. Our results indicate that the smallest percentile enabling full efficiency verifies $p(3) = 50\%$, $p(4) = 25\%$, $p(5) = 5\%$. Thus, for $\zeta = 100$, by extrapolating the observed speed of increase of efficiency as g increases, we conjecture that $p(6) \approx 1\%$, $p(7) < 1\%$, and $p(g) \ll 1\%$ for $g \geq 8$. However, the corresponding cardinal number of \mathcal{S} (same as the number of penultimate histograms to test to compute a single most likely path) can be estimated as $\text{card}(\mathcal{S}) \approx 5 \times 10^4, 3.2 \times 10^5, 5 \times 10^6, 10^8$ for $g = 3, 4, 5, 6$, respectively. Moreover, $\text{card}(\mathcal{S})$ seems to still increase at an exponential speed for $g \geq 7$. This is reflected by a significant

Table 4. Mean runtime per node in the parallel executions of the most likely path search with the p -quantile algorithm for the generation of \mathcal{S} .

p	100%	75%	50%	25%	10%	5%	1%
$g = 3$	3 min	2.5 min	2 min	1.5 min	1 min	1 min	1 min
$g = 4$	2 hrs	1.8 hrs	1 hr	7 min	3 min	2.5 min	1.5 min
$g = 5$	3.2 hrs	2.2 hrs	1 hr	38 min	32 min	30 min	26 min

Table 5. The most likely path H^* connecting $H = H_1$ and $G = H_6$ for $g = 7$ genotypes computed with the p -quantile algorithm. -

steps	histogram entries						
H_1	0.600	0.100	0.100	0.050	0.050	0.050	0.050
H_2	0.510	0.101	0.101	0.070	0.072	0.096	0.050
H_3	0.383	0.112	0.111	0.084	0.085	0.161	0.064
H_4	0.266	0.114	0.114	0.094	0.095	0.239	0.078
H_5	0.170	0.110	0.110	0.100	0.100	0.320	0.090
H_6	0.100	0.100	0.100	0.100	0.100	0.400	0.100

We display each histogram H_i on the i -th row of the table.

increase in computational time as g increases reported in Table 4. Therefore, this rules out using the p -quantiles algorithm for $g \geq 9$, even with efficient parallelization over 20 nodes.

4.4.2. Results for $g = 7$ and $g = 8$ genotypes

For a fixed g , the efficiency of the p -quantile algorithm is an increasing function of p , since the size of penultimate histograms, \mathcal{S} , is an increasing sequence, as p increases. Therefore, we can also define a *relative efficiency*, where we compare the most likely paths computed using two different sets of penultimate histograms, \mathcal{S}_{p_1} and \mathcal{S}_{p_2} , corresponding to two different percentiles $p_2 > p_1$. The most likely path computed with a higher percentile will always have a lower or equal cost compared to the most likely path computed with a lower percentile because $\mathcal{S}_{p_1} \subset \mathcal{S}_{p_2}$. Since the computation of the most likely paths for $g = 7, 8$ genotypes is much more expensive compared to $g \leq 6$, we analyze the relative efficiency for $g = 7, 8$ for only one pair of (H, G) and only two percentiles $p_1 = 10\%$ and $p_2 = 20\%$. We consider the mutation rate m , the mutation matrix Q , and growth factors F as discussed in Section B. The pairs of histograms (H, G) are presented below.

Simulations with $g = 7$ genotypes::

$$H = [0.6, 0.1, 0.1, 0.05, 0.05, 0.05, 0.05], \quad G = [0.1, 0.1, 0.1, 0.1, 0.1, 0.4, 0.1].$$

Simulations with $g = 8$ genotypes::

$$H = [0.5, 0.1, 0.1, 0.1, 0.05, 0.05, 0.05, 0.05], \quad G = [0.05, 0.05, 0.1, 0.05, 0.05, 0.1, 0.5, 0.1].$$

The discretization level for \mathcal{S} is chosen to be $\zeta = 100$ and $\zeta = 50$ for $g = 7$ and $g = 8$, respectively. The p -quantile algorithm for constructing \mathcal{S} results in the same most likely path for both $p_1 = 10\%$ and $p_2 = 20\%$ and both $g = 7, 8$. Thus, we conjecture that the efficiency of the p -quantile algorithm is close to one for $p = 10\%$ and $g = 7, 8$. The corresponding costs and lengths of the most likely paths for $g = 7, 8$ are $\text{cost}(H^*, g = 7) = 0.0264$, $\text{cost}(H^*, g = 8) = 0.0534$ and $T_{g=7}^* = 6$, $T_{g=8}^* = 7$. Most likely paths for $g = 7$ and $g = 8$ are presented in Tables 5 and 6, respectively.

For $g = 7$ and $g = 8$ computational tasks may not be distributed evenly over all parallel nodes due to the requirement to check dynamically whether a particular histogram belongs to

Table 6. The most likely path H^* connecting $H = H_1$ and $G = H_7$ for $g = 8$ genotypes computed with the p -quantile algorithm.

steps	histogram entries							
H_1	0.500	0.100	0.100	0.100	0.050	0.050	0.050	0.050
H_2	0.405	0.103	0.082	0.115	0.040	0.063	0.103	0.089
H_3	0.301	0.101	0.095	0.112	0.046	0.077	0.171	0.097
H_4	0.209	0.092	0.104	0.102	0.051	0.089	0.251	0.102
H_5	0.137	0.080	0.107	0.087	0.053	0.096	0.335	0.105
H_6	0.085	0.065	0.105	0.069	0.052	0.100	0.420	0.104
H_7	0.050	0.050	0.100	0.050	0.050	0.100	0.500	0.100

We display each histogram H_i on the i -th row of the table.

the set of penultimate histograms. Therefore, parallelization depends on how the histograms in \mathcal{S} are distributed over the nodes. Typically, the p -quantile is not parallelized well for large g , with only a few (between one and five) of the nodes performing the majority of computations.

We conclude that the p -quantile algorithm is feasible and produces adequate results $g \leq 8$, but becomes quite computationally expensive for $g \geq 7$. However, we also conclude that for $g \geq 9$ this strategy is not computationally feasible. Thus, we present an alternative geodesic shooting algorithm based on gradient descent.

5. Gradient descent for reverse geodesic shooting

Below, we describe an alternative strategy based on geodesic shooting.

5.1. Background: reverse geodesic shooting on riemannian manifolds

Let \mathcal{S} be a smooth Riemannian manifold of dimension k . Let us fix the time interval $[0, T]$, as well as two distinct points $x, y \in \mathcal{S}$. As discussed at the end of 3.3, to seek a Riemannian geodesic $f_t \in \mathcal{S}$ connecting x to y , one can solve—under the constraints $f_0 = x$ and $f_T = y$ —the Jacobi second-order differential equation

$$\left[\frac{\partial}{\partial f_t} - \frac{d}{dt} \frac{\partial}{\partial f'_t} \right] \text{Cost}(f_t, f'_t) = 0, \quad (26)$$

where $\text{Cost}(f_t, f'_t) = \langle f'_t, \gamma_{f_t} f'_t \rangle$ and γ_z is the quadratic form defined by the Riemann metric at $z \in \mathcal{S}$. The Jacobi ODE satisfied by f_t is then of the form

$$f''_t = \Gamma(f'_t, f_t) \quad \text{for } 0 < t < T, \quad (27)$$

where $\Gamma(u, v)$ is a fixed smooth function of $u \in \mathcal{S}, v \in \mathbb{R}^k$, easily derived from the Riemannian quadratic form γ_z .

Classical reverse geodesic shooting proceeds as follows: Let V_y be the tangent space to \mathcal{S} at point y . Fix any vector $v \in V_y$. Then solve the ODE (27) in *reverse time* $t < T$ after initialization by setting $f_T = y$ and $f'_T = v$. Then $f_t = F(t, y, v)$ is a smooth function of (t, y, v) . For fixed x and y , one seeks a vector $v \in V_y$ minimizing the Riemannian distance $\text{dist}(x, f_0)$ between the prescribed initial x and $f_0 = F(0, y, v)$. This search then proceeds by *numerical gradient descent* in v within the fixed tangent vector space V_y .

5.2. Gradient descent for geodesic shooting in the space of histograms

Given two interior histograms $H, G \in \mathcal{H}^+$, we now outline the gradient descent algorithm to compute the most likely path \mathbf{H}^* connecting H and G . The unknown optimal number of steps T^* has to be determined as well. We will successively explore the integer values $T = 3, 4, \dots, T_{\max}$ with the upper bound on the path's length $T_{\max} \leq 5g$. Let \mathcal{S} be the set of all histograms $Z \in \mathcal{H}^+$ such that $\min_{j=1, \dots, g} Z(j) \geq 0.005$. Consider a fixed $T \in [3, T_{\max}]$. Then for any $Z \in \mathcal{S}$ the geodesic shooting algorithm (see 4.1) computes the *open ended* reverse time geodesic $\{\theta_1 = G, \theta_2 = Z, \dots, \theta_t, \dots, \theta_{T-1}\}$ by the recursion $\theta_t = \chi(\theta_{t-1}, \theta_{t-2})$, iterated for $t \geq 2$ as long as θ_t is not a boundary histogram. Then the forward time path $H_t = \theta_{T-t+1}$, $t = 2, \dots, T \leq T_{\max}$, is a geodesic connecting histograms $H_2 = \theta_{T-1}$ with $H_T = G$ in $T - 1$ steps. This path also verifies $H_{T-1} = Z$. We can extend this path to $t = 1$ by setting $H_1 = H$. This adds a single first step from $H_1 = H$ to $H_2 \equiv \theta_{T-1}$ with an additional cost $C(H, H_2)$. This extension defines an approximate geodesic \mathbf{H} connecting H to G in T steps with penultimate histogram $H_{T-1} = Z$. This approximate geodesic is completely determined by Z and T . The total cost of this geodesic is given by

$$\lambda(\mathbf{H}) = C(H, H_2) + \sum_{t=2, \dots, T-1} C(H_t, H_{t+1}),$$

where $C(y, z)$ is the one-step cost given by (13). Recall that $C(y, z)$ is a smooth function of interior histograms y and z . Moreover, one has $H_t = \theta_{T-t+1}$ for $t \geq 2$, where θ_t is the smooth function of Z, G . Hence, the cost $\lambda(\mathbf{H})$ can be treated as a smooth function of the penultimate histogram, Z , if we consider H, G , and T fixed. One can compute the gradient of the total cost $W(Z) = \partial_Z \lambda(\mathbf{H}) \in \mathbb{R}^g$ with explicit formulas summarized in the next section. Thus, we can use the gradient descent algorithm to compute the optimal path \mathbf{H}^* . The gradient descent algorithm results in a sequence of penultimate histograms $Z_1, Z_2, \dots, Z_n, \dots$ and corresponding paths $\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_n, \dots$. We initialize the gradient decent algorithm at random, such that for $Z_1 \in \mathcal{S}$ the path $\mathbf{H}_1 = [H, H_{(1),2}, \dots, H_{(1),T-1} = Z_1, H_{(1),T} = G]$ consists of interior histograms. For $n \geq 2$, gradient descent algorithm computes

$$Y = Z_{n-1} - \alpha_n W(Z_{n-1}), \quad s(Y) = \sum_{j=1}^g Y(j), \quad \text{and} \quad Z_{n+1} = Y/s(Y),$$

where Z_{n-1} is the penultimate histogram from the previous step, computed together with the cost of the path $\lambda(\mathbf{H}_{n-1})$ and the gradient $W(Z_{n-1})$. The gain parameter $\alpha_n > 0$ needs to be chosen small enough to guarantee $\min_j Y(j) > 0$ and also guarantee that the path \mathbf{H}_n consists of interior histograms. In addition, we use the Armijo line search [?] to calibrate the step size α_n . The gradient descent algorithm is terminated at step n if either $n = 500$ or if the cost $\lambda(\mathbf{H}_n)$ stopped decreasing. We perform the gradient decent for all values of $T \in [3, T_{\max}]$ and select the most likely path \mathbf{H}^* connecting H and G as the path with the minimal total cost with respect to T .

5.3. Gradients of the rate functions

Before we present our results, we provide the derivations of the expressions used for our gradient descent algorithm outlined in the former section.

5.3.1. Gradient of rate function for reverse geodesics. Any geodesics (represented in reverse time) $\theta = [\theta_1 \equiv G, \theta_2 \equiv Z, \dots, \theta_T \equiv H]$ is fully determined by the final and penultimate histograms G and Z , since for $t \geq 3$ θ_t is given by a recursive relationship $\theta_t = \chi(\theta_{t-1}, \theta_{t-2})$ in (22). The trajectory cost is given by $\lambda(\theta) = \sum_{t=1}^{T-1} C(\theta_{t+1}, \theta_t)$ where $C(\cdot, \cdot)$ is the one-step cost in (15).

Next, we consider the number of steps, T , fixed and derive the gradient $D_Z \lambda(\theta)$ with respect to the penultimate histogram Z . Using the chain rule

$$D_Z \lambda(\theta) = \sum_{t=1}^{T-1} \left[\nabla_x C(x, y) \Big|_{(x,y)=(\theta_{t+1}, \theta_t)} D_Z \theta_{t+1} + \nabla_y C(x, y) \Big|_{(x,y)=(\theta_{t+1}, \theta_t)} D_Z \theta_t \right], \quad (28)$$

where $\nabla_x C(x, y)$ and $\nabla_y C(x, y)$ are row vectors and $D_Z \theta_{t+1}$ and $D_Z \theta_t$ are matrices with $(D_Z \theta_t)_{ij} = \partial \theta_t(i) / \partial z(j)$. Then $D_Z \lambda(\theta)$ is a column vector.

We can express the derivative $D_Z \theta_t$ as

$$D_Z \theta_t = D_x \chi(x, y) \Big|_{(x,y)=(\theta_{t-1}, \theta_{t-2})} D_Z \theta_{t-1} + D_y \chi(x, y) \Big|_{(x,y)=(\theta_{t-1}, \theta_{t-2})} D_Z \theta_{t-2}, \quad (29)$$

where both $D_x \chi(x, y)$ and $D_y \chi(x, y)$ are matrices. For $t = 1, 2$, $D_Z \theta_1 \equiv D_Z G = 0$ and $D_Z \theta_2 \equiv D_Z Z = I$ (the identity matrix). Therefore, formula (29) can be used to compute $D_Z \theta_t$ for $t > 2$. After that, formula (28) can be used to compute the gradient $D_Z \lambda(\theta)$. Next, we need to derive expressions for partial derivatives of $C(x, y)$ and $\chi(x, y)$, where x and y are two arbitrary internal histograms.

5.3.2. Gradient of one-step cost. The one-step cost $C(H, G)$ from H to G is given by

$$C(H, G) = KL(G, \Phi) + m \sum_{j \neq k} F_j H(j) Q_{j,k} (1 - U_k / U_j),$$

where

$$\Phi(j) = \frac{F_j H(j)}{\langle F, H \rangle}, \quad KL(G, \Phi) = \sum_j G(j) \log(G(j) / \Phi(j)), \quad U_j = \exp \left(\frac{G(j)}{F_j H(j)} \right).$$

Since H is a histogram normalized to one, its entries are *not* independent and we can express $H(g) = 1 - \sum_{j=1 \dots g-1} H(j)$. Thus, we compute the partial derivatives of Φ above with respect to $H(j)$ for $j < g$. We obtain the following expressions

$$\begin{aligned} \partial_{H(j)} \Phi(j) &= \frac{F_j \langle F, H \rangle - F_j H(j) (F_j - F_g)}{\langle F, H \rangle^2} & \text{for } j < g, \\ \partial_{H(k)} \Phi(j) &= \frac{F_j H(j) (F_g - F_k)}{\langle F, H \rangle^2} & \text{for } j, k < g, j \neq k, \\ \partial_{H(j)} \Phi(g) &= \frac{-F_g \langle F, H \rangle - F_g H(g) (F_j - F_g)}{\langle F, H \rangle^2} & \text{for } j < g. \end{aligned}$$

Derivatives of KL divergence can be computed as follows

$$\partial_{H(j)} KL = - \sum_k \frac{G(k)}{\Phi(k)} \partial_{H(j)} \Phi(k), \quad \partial_{G(j)} KL = \log \frac{G(j)}{\Phi(j)} - \log \frac{G(g)}{\Phi(g)}.$$

We can also obtain explicit expressions for the derivatives of U_j :

$$\partial_{H(j)} U_j = -U_j G(j) / F_j H(j)^2 \quad \text{and} \quad \partial_{G(j)} U_j = U_j / F_j H(j) \quad \text{for } j < g,$$

and

$$\partial_{H(j)} U_g = U_g G(g) / F_g H(g)^2, \quad \partial_{G(j)} U_g = -U_g / F_g H(g).$$

The differential $D_H U$ is a matrix of partial derivatives $(D_H U)_{j,k} = \partial_{H(k)} U_j$. Denote

$$\rho = \rho(H, G) = m \sum_{j \neq k} F_j H(j) Q_{j,k} U_k / U_j.$$

For $s < g$, the partial derivatives of ρ with respect to $H(s)$ and $G(s)$ are given by

$$\begin{aligned} \partial_{H(s)} \rho(H, G) = m & \left[F_g Q_{g,s} \left(-\frac{U_s}{U_g} + H(g) \frac{(D_H U)_{s,s} U_g - U_s (D_H U)_{g,s}}{U_g^2} \right) \right. \\ & + F_s Q_{s,g} \left(\frac{U_g}{U_s} + H(s) \frac{(D_H U)_{g,s} U_s - U_g (D_H U)_{s,s}}{U_s^2} \right) \\ & + \sum_{\substack{k=1, \dots, g-2 \\ k \neq s}} \left(F_s Q_{s,k} \left(\frac{U_k}{U_s} - \frac{H(s) U_k (D_H U)_{s,s}}{U_s^2} \right) + m F_k H(k) Q_{k,s} \frac{(D_H U)_{s,s}}{U_k} \right) \\ & + \sum_{\substack{k=1, \dots, g-2 \\ k \neq s}} \left(-F_g Q_{g,k} U_k \left(\frac{1}{U_g} + \frac{H(g) (D_H U)_{g,s}}{U_g^2} \right) \right. \\ & \left. \left. + m F_k H(k) Q_{k,g} \frac{(D_H U)_{g,s}}{U_k} \right) \right], \end{aligned}$$

and

$$\begin{aligned} \partial_{G(s)} \rho(H, G) = m & \left[\sum_{\substack{k=1, \dots, g-1 \\ k \neq s}} \left(-Q_{s,k} \frac{U_k}{U_s} + F_k H(k) Q_{k,s} \frac{U_s}{U_k F_s H(s)} \right) \right. \\ & \left. + \sum_{k=1}^{g-1} \left(Q_{g,k} \frac{U_k}{U_g} - F_k H(k) Q_{k,g} \frac{U_g}{U_k F_g H(g)} \right) \right]. \end{aligned}$$

The formulas above yield the partial derivatives of the cost $C(H, G)$ with respect to $H(j)$ and $G(j)$ for $j < g$,

$$\begin{aligned} \partial_{H(j)} C(H, G) &= \partial_{H(j)} KL - \partial_{H(j)} \rho(H, G), \\ \partial_{G(j)} C(H, G) &= \partial_{G(j)} KL - \partial_{G(j)} \rho(H, G). \end{aligned}$$

These last two formulas then provide the gradients $\nabla_H C(H, G)$ and $\nabla_G C(H, G)$.

5.3.3. Gradient of reverse geodesic. As discussed previously, any geodesics $\{\theta_T, \theta_{T-1}, \theta_1\}$ satisfies the recursive relationship given by $\theta_t = \chi(\theta_{t-1}, \theta_{t-2})$. Thus, we consider partial derivatives of the vector-valued function $x = \chi(y, z)$, where the function $\chi(y, z)$ is defined by the following expressions

$$X_s = \frac{y_s}{F_s} \exp \left(\frac{F_s}{\langle F, y \rangle} - \frac{z_s}{y_s} \right), \quad Y_s = \frac{X_s}{\sum_t X_t},$$

$$\begin{aligned}
e_{s,k} &= \exp\left(-\frac{y_s}{F_s Y_s} + \frac{y_k}{F_k Y_k}\right), \\
\alpha_s &= \sum_{k \neq s} \left(Q_{s,k} e_{s,k} - \frac{F_k X_k}{F_k X_s} Q_{k,s} e_{k,s} \right), \\
f_{s,k} &= \exp\left(-\frac{z_s}{F_s y_s} + \frac{z_k}{F_k y_k}\right), \\
\beta_s &= F_s \sum_k Q_{s,k} - \left(F_s + \frac{z_s}{y_s} \sum_k f_{s,k} Q_{s,k} - \frac{z_s}{F_s y_s^2} \sum_k F_k y_k Q_{k,s} f_{k,s} \right), \\
w_s &= \alpha_s + \beta_s - \langle Y, \alpha + \beta \rangle, \\
x_s &= Y_s + m Y_s w_s.
\end{aligned}$$

For any vector valued function $V = V(y, z) \in \mathbb{R}^g$ differentials with respect to $y, z \in \mathbb{R}^g$ are $g \times g$ matrices with entries $(D_y V)_{ij} = \partial_{y(j)} V(y, z)(i)$ and $(D_z V)_{ij} = \partial_{z(j)} V(y, z)(i)$. Similarly, the differentials $D_y e$, $D_z e$, $D_y f$, and $D_z f$ of the matrices e and f are $g \times g \times g$ tensors, with coefficients denoted $(D_y e)_{s,k,q} = \partial_{y(q)} e_{s,k}$, with similar notations for other tensor differentials.

From the expression for X_s we obtain

$$\begin{aligned}
(D_y X)_{s,s} &= \exp\left(\frac{F_s}{\langle F, y \rangle} - \frac{z_s}{y_s}\right) \left(\frac{1}{F_s} + \frac{z_s}{F_s y_s} - \frac{y_s(F_s - F_g)}{\langle F, y \rangle^2} \right), \\
(D_z X)_{s,s} &= -\frac{1}{F_s} \exp\left(\frac{F_s}{\langle F, y \rangle} - \frac{z_s}{y_s}\right), \\
(D_z X)_{s,k} &= 0, \quad (D_y X)_{s,k} = \frac{y_s(F_g - F_k)}{\langle F, y \rangle^2} \exp\left(\frac{F_s}{\langle F, y \rangle} - \frac{z_s}{y_s}\right) \quad \text{for } s \neq k.
\end{aligned}$$

Let $S = \sum_s X_s \in \mathbb{R}$. Then one has

$$\begin{aligned}
(D_z S)_k &= (D_z X)_{k,k} \\
(D_y S)_k &= \exp\left(\frac{F_k}{\langle F, y \rangle} - \frac{z_k}{y_k}\right) \left(\frac{1}{F_k} + \frac{z_k}{F_k y_k} - \frac{y_k(F_k - F_g)}{\langle F, y \rangle^2} \right) \\
&\quad + \sum_{\substack{s=1, \dots, g \\ s \neq k}} \exp\left(\frac{F_s}{\langle F, y \rangle} - \frac{z_s}{y_s}\right) \frac{y_s(F_g - F_k)}{\langle F, y \rangle^2}.
\end{aligned}$$

Then we can compute

$$(D_y Y)_{s,k} = \frac{S(D_y X)_{s,k} - X_s(D_y S)_k}{S^2} \quad \text{and} \quad (D_z Y)_{s,k} = \frac{S(D_z X)_{s,k} - X_s(D_z S)_k}{S^2}.$$

Next, differentiating $e_{s,k}$ we obtain for $q \neq s \neq k$,

$$\begin{aligned}
(D_y e)_{s,k,s} &= e_{s,k} \left(\frac{y_s(D_y Y)_{s,s} - Y_s}{Y_s^2} - \frac{y_k(D_y Y)_{k,s}}{F_k Y_k^2} \right), \\
(D_y e)_{s,k,k} &= e_{s,k} \left(\frac{y_s(D_y Y)_{s,k}}{F_s Y_s^2} + \frac{Y_k - y_k(D_y Y)_{k,k}}{Y_k^2} \right),
\end{aligned}$$

$$(D_y e)_{s,k,q} = e_{s,k} \left(\frac{y_s (D_y Y)_{s,q}}{F_s Y_s^2} - \frac{y_k (D_y Y)_{k,q}}{F_k Y_k^2} \right),$$

$$(D_z e)_{s,k,q} = e_{s,k} \left(\frac{y_s}{F_s Y_s^2} \cdot (D_z Y)_{s,q} - \frac{y_k}{F_k Y_k^2} \cdot (D_z Y)_{k,q} \right).$$

Differentiating α we obtain for $q < g$

$$(D_y \alpha)_{s,q} = \sum_{k \neq q} Q_{s,k} (D_y e)_{s,k,q} - \sum_{k \neq s} \frac{F_k Q_{k,s}}{F_s} \left(e_{k,s} \frac{(D_y X)_{k,q} X_s - X_k (D_y X)_{s,q}}{X_s^2} + \frac{X_k (D_y e)_{k,s,q}}{X_s} \right)$$

and for $s < g$

$$(D_z \alpha)_{s,s} = \sum_{k \neq s} \left(Q_{s,k} (D_z e)_{s,k,s} - \frac{F_k X_k Q_{k,s}}{F_s} \frac{(D_z e)_{k,s,s} X_s - e_{k,s} (D_z X)_{s,s}}{X_s^2} \right).$$

Next, for $s = 1, \dots, g$, $q < g$ and $s \neq q$,

$$(D_z \alpha)_{s,q} = Q_{s,q} (D_z e)_{s,q,q} - \frac{F_q Q_{q,s}}{F_s X_s} (e_{q,s} (D_z X)_{q,q} + X_q (D_z e)_{q,s,q})$$

$$+ \sum_{k \neq s, k \neq q} \left(Q_{s,k} (D_z e)_{s,k,q} - \frac{F_k X_k}{F_s X_s} Q_{k,s} (D_z e)_{k,s,q} \right).$$

The expression for $f_{s,k}$ implies that for $s < g$, $k = 1, \dots, g$,

$$(D_y f)_{s,k,s} = \frac{f_{s,k} z_s}{F_s y_s^2} \quad \text{and} \quad (D_z f)_{s,k,s} = -\frac{f_{s,k}}{F_s y_s},$$

and for $s = 1, \dots, g$, $k < g$, $q \neq s$, $q \neq k$,

$$(D_y f)_{s,k,k} = -\frac{f_{s,k} z_k}{F_k y_k^2}, \quad (D_z f)_{s,k,k} = \frac{f_{s,k}}{F_k y_k}, \quad \text{and} \quad (D_y f)_{s,k,q} = 0.$$

The formula for β implies that for $s < g$,

$$(D_y \beta)_{s,s} = \frac{z_s}{y_s^2} \sum_k f_{s,k} Q_{s,k} - \left(F_s + \frac{z_s}{y_s} \right) \sum_{k \neq s} (D_y f)_{s,k,s} Q_{s,k}$$

$$+ \frac{2z_s}{F_s y_s^3} \sum_k F_k y_k Q_{k,s} f_{k,s} - \frac{z_s}{F_s y_s^2} \sum_{k \neq s} F_k y_k Q_{k,s} (D_y f)_{k,s,s},$$

$$(D_z \beta)_{s,s} = -\frac{1}{y_s} \sum_k f_{s,k} Q_{s,k} - \left(F_s + \frac{z_s}{y_s} \right) \sum_k Q_{s,k} (D_z f)_{s,k,s}$$

$$- \frac{1}{F_s y_s^2} \sum_k F_k y_k Q_{k,s} f_{k,s} - \frac{z_s}{F_s y_s^2} \sum_k F_k y_k Q_{k,s} (D_z f)_{k,s,s}.$$

For $s = 1, \dots, g$, $q < g$ and $s \neq q$, we have

$$(D_y \beta)_{s,q} = -\left(F_s + \frac{z_s}{y_s} \right) Q_{s,q} (D_y f)_{s,q,q} - \frac{z_s F_q y_q Q_{q,s} (D_y f)_{q,s,q}}{F_s y_s^2} - \frac{z_s F_q Q_{q,s} f_{q,s}}{F_s y_s^2},$$

$$(D_z \beta)_{s,q} = -\left(F_s + \frac{z_s}{y_s} \right) \sum_{k \neq s} Q_{s,k} (D_z f)_{s,k,q} - \frac{z_s}{F_s y_s^2} \sum_{k \neq s} F_k y_k Q_{k,s} (D_z f)_{k,s,q}.$$

The formula for w implies that for $s < g, k < g$

$$\begin{aligned}(D_y w)_{s,k} &= (D_y \alpha)_{s,k} + (D_y \beta)_{s,k} - (D_y S)_k, \\ (D_z w)_{s,k} &= (D_z \alpha)_{s,k} + (D_z \beta)_{s,k} - (D_z S)_k.\end{aligned}$$

Finally, from $x_s = Y_s + mY_s w_s$, we obtain

$$\begin{aligned}(D_y x)_{s,k} &= (D_y Y)_{s,k}(1 + mw_s) + mY_s(D_y w)_{s,k}, \\ (D_z x)_{s,k} &= (D_z Y)_{s,k}(1 + mw_s) + mY_s(D_z w)_{s,k}.\end{aligned}$$

Since $x = \chi(y, z)$ the last two formulas provide gradients of $\chi(y, z)$ with respect to y and z .

5.4. Numerical results

For $g = 8$, we tested the gradient descent algorithm for the same pair H, G used in [Section 4.4.2](#) and parameter values in [Section B](#). For $g = 8$, the computational time for the p -quantile algorithm discussed in [Section 4.4.2](#) is approximately one hour. The optimal penultimate histogram and the total trajectory cost computed given by the p -quantile algorithm are given by the row H_6 in [6](#) and $\tilde{\lambda} = 0.0534$, respectively.

The gradient descent algorithm yields the most likely path with a smaller overall cost $\lambda^* = 0.0343$ and the penultimate histogram

$$Z^* = [0.0841, 0.0651, 0.1129, 0.0664, 0.0564, 0.0999, 0.4188, 0.0963].$$

Moreover, the computational time using the gradient descent algorithm was drastically reduced to 20 s on a single computational node. Therefore, the gradient decent algorithm for computing the most likely path is superior to the p -quantile approach discussed earlier in this article.

Due to a significant reduction in computational time using the gradient descent algorithm, we can apply this approach for $g = 10$ genotypes. Parameters are given in [Section B](#). The initial and the target histograms were chosen as

$$\begin{aligned}H &= [0.45, 0.10, 0.10, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05], \\ G &= [0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.10, 0.45, 0.10].\end{aligned}$$

The gradient descent algorithm yields the best geodesic $H^* = \{H_1 = H, \dots, H_9 = G\}$ connecting H to G in $T^* = 9$ steps, with the total trajectory cost $\lambda(H^*) = 0.0259$ and penultimate histogram

$$Z^* = H_8 = [0.0726, 0.0606, 0.0625, 0.0567, 0.0572, 0.0573, 0.0559, 0.0997, 0.3810, 0.0974].$$

The total computing time was around 40 s on one single node. Here, we present the analysis of convergence for $T^* = 9$.

We also studied the rate of convergence of norm $\|H - \theta_{T^*-1}\|^2$ and the total cost of the trajectory with respect to the optimization step, n . We expect to observe linear convergence since we consider a standard (first-order) gradient descent scheme. We observe that the norm and the cost did not change drastically after approximately $n = 18$ for the experiments carried out in this section, showcasing a quick convergence to a good solution for the considered problem (detailed results are reported in [\[26\]](#)). Since the convergence of gradient descent is in general not independent of the problem dimension, we expect to observe a slower rate at

larger values for g . Accelerated gradient descent schemes (with momentum) or quasi-Newton methods can serve as a remedy. However, given the overall small dimension ($g \ll 1000$), we do not think higher-order optimization algorithms are required.

The gradient descent algorithm is considerably faster compared to a more straightforward p -quantile discretization approach. In particular, the computing time for $g = 10$ only doubles compared with the computing time for $g = 8$. Thus, the gradient descent algorithm can be successfully used to analyze the most likely paths for a relatively large number of genotypes. This work will be carried out in a successive paper where we analyze realistic biological scenarios of rare events.

The efficiency and performance of the gradient descent algorithm depend on the initial penultimate histogram, Z_1 , and selecting an adequate starting penultimate histogram is quite crucial to get the correct most likely path connecting H and G . Similar to other applications of the gradient descent algorithm to nonlinear problems, the optimization search can converge to a local minimum. In addition, the gradient descent algorithm is only applicable if the trajectory generated by Z_1 consists of interior histograms. Therefore, we performed studies of how random selection of Z_1 affects the performance of the gradient descent algorithm. In particular, we generated $K = 154$ initial penultimate histograms $\{Z_1^{(k)}, k = 1, \dots, K\}$ and after fixing the number of steps the optimal value ($T^* = 7$ and $T^* = 9$ for $g = 8$ and $g = 10$, respectively), we performed gradient decent search for each penultimate histogram. For $g = 8$, 131 paths out of 154 resulted in interior paths. Moreover, approximately 50 paths had a trajectory cost essentially equal to 0.03425, which is approximately the optimal cost for $g = 8$. This corresponds to a success rate of $\approx 38\%$ for random initializations of the gradient descent approach for $g = 8$. For $g = 10$, only 52 out of 154 random choices of Z_1 resulted in paths consisting of interior histograms. Moreover, among these 52 paths, 25 paths had costs in the range $[0.0216, 0.0235]$ and only five had the approximate lowest cost of 0.0216. This implies that it is necessary to perform many realizations of the gradient descent algorithm with many values of the penultimate histogram. It is possible to combine a coarse discretization of the space of interior histograms \mathcal{H} with the gradient descent algorithm or to find a good starting point by performing a coarse greedy search to find a good initial point.

6. Conclusions

In this article, we discuss the large deviations theory for Markov Chains modeling genetic evolution for bacterial populations. In particular, these models describe the long-term laboratory *E. coli* experiments where cells undergo daily growth, mutations, and dilutions. It has been demonstrated that such long-term evolutionary experiments often lead to the emergence of new bacterial genotypes. Thus, the Markov chains discussed in this article are aimed at modeling the stochastic dynamics of population histograms of genotype frequencies. We outlined the main theoretical results from a previous paper [24, 25]. These theoretical results lay the mathematical foundation for deriving the cost function for evolutionary trajectories connecting two arbitrary histograms. In particular, we focus on rare events and the large deviation analysis of trajectories when one of the emergent bacterial sub-populations does not have the highest fitness (i.e., not the dominant genotype in the population). Such trajectories correspond to the cost-minimizing paths and have very small probabilities, which cannot be computed directly by, for instance, ensemble simulations.

We present several algorithms for computing the most likely trajectories connecting the initial and the final histograms. The computation is performed “in reverse time” and the most likely trajectories are completely determined by the final and penultimate histograms. Since the penultimate histogram is not known, this leads to an optimization problem with respect to the penultimate histogram. We present several algorithms for computing the most likely trajectories connecting the initial and the final histograms. The computation is performed “in reverse time” and the most likely trajectories are completely determined by the final and penultimate histograms. Since the penultimate histogram is not known, this leads to an optimization problem with respect to the penultimate histogram. We compare and contrast the straightforward and the p -quantiles search algorithms in [Section 4](#). These algorithms are based on a relatively straightforward optimization problem, where the set of candidate penultimate histograms is constructed with a relatively coarse discretization. We discuss the efficiency of the p -quantile search algorithm and demonstrate its applicability for the number of genotypes $g \leq 8$. and the p -quantile search algorithms in [Section 4](#). These algorithms are based on a relatively straightforward optimization problem, where the set of candidate penultimate histograms is constructed with a relatively coarse discretization. We discuss the efficiency of the p -quantile search algorithms and demonstrate its applicability for the number of genotypes $g \leq 8$.

To handle problems with a larger number of possible genotypes in bacterial populations, we develop a more efficient gradient descent algorithm in [Section 5](#). In particular, we derive explicit expressions for the gradient of the cost function with respect to the penultimate histogram. We demonstrate that this algorithm is easily computationally applicable to problems with $g \leq 10$ genotypes. Moreover, we also estimate that this algorithm is potentially applicable for problems with up to $g = 20$ genotypes. Since the corresponding optimization problem is nonconvex, one potential drawback of the gradient descent algorithm is that it might converge to a local minimum, as discussed at the end of [Section 5.4](#). Therefore, for problems with $g > 10$ genotypes, one probably has to combine the gradient descent algorithm with the p -quantiles approach, so that the starting penultimate histogram in the gradient descent algorithm is selected near an optimal point. The computational framework developed in this article stands ready to tackle interesting biological problems such as the emergence of genotypes with lower fitness but with a higher degree of adaptation. This involves considering a mutation matrix Q with non-uniform mutation rates. These experiments will be carried out in a consecutive paper.

Numerical algorithms presented in this article can be potentially used for two types of applications—(i) reconstructing the most likely path connecting the initial and the final histograms and (ii) performing the importance sampling MC simulations in path space and, thus, computing the small probabilities of reaching the target histogram. Both of these applications were addressed in [\[26\]](#). In practice, it is not possible to determine the genetic composition of the bacterial population during each generation, since these experiments often run for tens of thousands of generations (see, e.g., [\[28\]](#)). Therefore, the detailed genetic evolutionary path can be reconstructed using the numerical technique presented in this article. The second application involves performing MC importance sampling simulations where the original Markov chain is modified so that all paths remain close to the most likely path connecting the initial and the final histograms. This enables computing accurate estimates of the actual probability of reaching the target histogram. We demonstrated that

our computational strategy allows addressing both applications in a practical setting with a relatively large number of genotypes $g \approx 20$.

Acknowledgments

Any opinions, findings, conclusions, or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the NSF. This work was completed in part with resources provided by the Research Computing Data Core at the University of Houston.

Disclosure statement

No potential competing interest was reported by the authors.

Funding

This work was partly supported by the National Science Foundation (NSF) through the grants DMS-2009923, DMS-2012825, DMS-2145845, and DMS-1903270.

A. summary of notation

Below, we summarize the notation used in this manuscript.

- $g \in \mathbb{N}$: number of genotypes in the bacterial population
- $N \in \mathbb{N}$: number of cells in the bacterial population
- $m \in \mathbb{R}$: mutation rate
- $H \in \mathbb{R}^g$: population histogram of bacterial frequencies; $H = [H(1), H(2), \dots, H(g)]$ with $H(j) \geq 0$ for $j = 1, \dots, g$ and $\sum_{j=1}^g H(j) = 1$
- $H(j) \in [0, 1]$: frequency of genotype j in the population
- $H_t \in \mathbb{R}^g$: population histogram of bacterial frequencies on day t
- $H_t(j) \in [0, 1]$: frequency of genotype j in the population on day t
- $[F_1, F_2, \dots, F_g]$: ordered genotype growth factors with $F_1 < F_2 < \dots < F_g$
- $\mathcal{H} \subset \mathbb{R}^g$: space of all histograms
- $\mathcal{H}^+ = \{H : H \in \mathcal{H} \text{ and } H(j) > 0 \text{ for all } j = 1, \dots, g\}$ is the set of all interior histograms
- $\mathbf{H} = \{H_1, \dots, H_T\}$: time-dependent path of length T (days) in the space \mathcal{H} connecting the initial histogram H_1 and the final histogram H_T
- Ω_T : space of all paths of length T
- Ω_T^+ : space of all interior paths of length T ; if $\mathbf{H} \in \Omega_T^+$ then $H_t(j) \geq \varepsilon$ for all $1 \leq t \leq T$ and $1 \leq j \leq g$ and for some $\varepsilon \ll 1$

B. growth factors and selective advantages

In all simulations in this article the mutation rate is $m = 10^{-8}$. The $g \times g$ mutant transfer matrix Q has all diagonal coefficients $Q_{i,i} = 0$, and all non-diagonal coefficients $Q_{i,j} = 1/g - 1$ for $i \neq j$.

For simulations with $g = 3, 4, 5$ genotypes growth factors $F = [F_1, \dots, F_g]$ are given by

- $g = 3$: $F = [200, 305.6, 377.7]$,
- $g = 4$: $F = [200, 305.6, 339.7, 377.7]$,
- $g = 5$: $F = [200, 305.6, 339.7, 358.2, 377.7]$.

For simulations with $g = 7, 8$, and 10 genotype growth factors are given by

- $g = 7$: $F = [200, 289.8, 305.6, 322.2, 339.7, 358.2, 377.7]$,
- $g = 8$: $F = [200, 274.8, 289.8, 305.6, 322.2, 339.7, 358.2, 377.7]$,
- $g = 10$: $F = [200, 200^{1.04}, 200^{1.05}, 200^{1.06}, 200^{1.07}, 200^{1.08}, 200^{1.09}, 200^{1.10}, 200^{1.11}, 200^{1.12}]$.

References

- [1] Lamrabet, O., Martin, M., Lenski, R. E., Schneider, D. (2019). Changes in intrinsic antibiotic susceptibility during a long-term evolution experiment with *Escherichia coli*. *mBio*. 10: 189–208.
- [2] Barrick, J. E., Kauth, M. R., Strelisoff, C. C., Lenski, R. E. (2010). *Escherichia coli* rpoB mutants have increased evolvability in proportion to their fitness defects. *Mol. Biol. Evol.* 27(6): 1338–1347. doi: [10.1093/molbev/msq024](https://doi.org/10.1093/molbev/msq024).
- [3] Woods, R. J., Barrick, J. E., Cooper, T. F., Shrestha, U., Kauth, M. R., Lenski, R. E. (2011). Second-order selection for evolvability in a large *Escherichia coli* population. *Science*. 331(6023): 1433–1436. doi: [10.1126/science.1198914](https://doi.org/10.1126/science.1198914).
- [4] Grant, N. A., Magid, A. A., Franklin, J., Dufour, Y., Lenski, R. E. (2021). Changes in cell size and shape during 50,000 generations of experimental evolution with *Escherichia coli*. *J. Bacteriol.* 203(10): e00469-20. doi: [10.1128/jb.00469-20](https://doi.org/10.1128/jb.00469-20).
- [5] Dembo, A., Zeitouni, O. (1998). *Large Deviations Techniques and Applications*. Berlin, Heidelberg: Springer.
- [6] Varadhan, S. R. S. (1984). Large deviations and applications. In *CBMS-NSF*, Vol. 46. Philadelphia, PA: SIAM.
- [7] Freidlin, M., Wentzell, A. (1998). *Random Perturbations of Dynamical Systems*, Vol. 260. Berlin, Heidelberg: Springer.
- [8] Azencott, R., Ruget, G. (1977). Mélanges d'équations différentielles et grands écarts à la loi des grands nombres. *Z Wahrscheinlichkeitstheorie Verw. Gebiete*. 38(1): 1–54. doi: [10.1007/BF00534169](https://doi.org/10.1007/BF00534169).
- [9] Azencott, R. (1980). Large deviations theory and applications, in *Probabilities at Saint-Flour VIII-1978*, Lect. Notes Math, H. P. L., 774, p. 1–176.: Springer.
- [10] Azencott, R., Freidlin, M. I., Varadhan, S. (2013). *Large Deviations at Saint-Flour, Springer Series "Probabilities at Saint-Flour"* Vol. VIII, New York: Springer.
- [11] Bouchet, F., Grafke, T., Tangarife, T., Vanden-Eijnden, E. (2016). Large deviations in fast–slow systems. *J. Stat. Phys.* 162(4): 793–812. doi: [10.1007/s10955-016-1449-4](https://doi.org/10.1007/s10955-016-1449-4).
- [12] Grafke, T., Schäfer, T., Vanden-Eijnden, E. (2017). *Long Term Effects of Small Random Perturbations on Dynamical Systems: Theoretical and Computational Tools*. New York, NY: Springer, pp. 17–55.
- [13] Grafke, T., Vanden-Eijnden, E. (2019). Numerical computation of rare events via large deviation theory. *Chaos*. 29: 63118.
- [14] Champagnat, N. (2006). A microscopic interpretation for adaptive dynamics trait substitution sequence models. *Stoch. Process. Appl.* 116(8): 1127–1160. doi: [10.1016/j.spa.2006.01.004](https://doi.org/10.1016/j.spa.2006.01.004).
- [15] Champagnat, N., Ferriere, R., Arous, G. B. (2001). *The Canonical Equation of Adaptive Dynamics: A Mathematical View*, *Selection* 2: 71–81.
- [16] Champagnat, N., Ferrière, R., Méléard, S. (2006). Unifying evolutionary dynamics: from individual stochastic processes to macroscopic models. *Theor. Popul. Biol.* 69(3): 297–321. doi: [10.1016/j.tpb.2005.10.004](https://doi.org/10.1016/j.tpb.2005.10.004).
- [17] Champagnat, N., Lambert, A. (2007). Evolution of discrete populations and the canonical diffusion of adaptive dynamics. *Ann. Appl. Prob.* 17: 102–155.
- [18] Zhang, Wei., Sehgal, V., Dinh, D. M., Azevedo, R. B. R., Cooper, T. F., Azencott, R. (2012). Estimation of the rate and effect of new beneficial mutations in asexual populations. *Theor. Popul. Biol.* 81(2): 168–178. doi: [10.1016/j.tpb.2011.11.005](https://doi.org/10.1016/j.tpb.2011.11.005).

- [19] Hegreness, M., Shores, N., Hartl, D., Kishony, Roy. (2006). An equivalence principle for the incorporation of favorable mutations in asexual populations. *Science*. 311(5767): 1615–1617. doi: [10.1126/science.1122469](https://doi.org/10.1126/science.1122469).
- [20] Moura De Sousa, J. A., Campos, P. R. A., Gordo, I. (2013). An ABC method for estimating the rate and distribution of effects of beneficial mutations. *Genome Biol. Evol.* 5(5): 794–806. doi: [10.1093/gbe/evt045](https://doi.org/10.1093/gbe/evt045).
- [21] Illingworth, C. J. R., Mustonen, V. (2012). A method to infer positive selection from marker dynamics in an asexual population. *Bioinformatics*. 28(6): 831–837. doi: [10.1093/bioinformatics/btr722](https://doi.org/10.1093/bioinformatics/btr722).
- [22] Simon, D. (2013). *Evolutionary Optimization Algorithms*. New York: John Wiley & Sons.
- [23] Ross, S. M. (1996). *Stochastic Processes*. New York: John Wiley & Sons.
- [24] Azencott, R., Geiger, B., Timofeyev, I. (2023). Large deviations analysis for stochastic models of bacterial evolution. arxiv:1811.10176
- [25] Geiger, B. J. (2017). Large deviations for dynamical systems with small noise. PhD thesis, University of Houston, Houston, TX. <https://hdl.handle.net/10657/4801>.
- [26] Su, Y. (2022) *Rare events simulation in bacterial genetic evolution models*, PhD thesis, University of Houston, <https://hdl.handle.net/10657/10734>.
- [27] Lenski, R. E., Rose, M. R., Simpson, S. C., Tadler, S. C. (1991). long-term experimental evolution in *Escherichia coli*. I. Adaptation and divergence during 2,000 generations. *Am. Natural.* 138(6): 1315–1341. doi: [10.1086/285289](https://doi.org/10.1086/285289).
- [28] Cooper, T. F., Rozen, D. E., Lenski, R. E. (2003). Parallel changes in gene expression after 20,000 generations of evolution in *Escherichia coli*. *Proc. Natl. Acad. Sci. USA*. 100(3): 1072–1077. doi: [10.1073/pnas.0334340100](https://doi.org/10.1073/pnas.0334340100).
- [29] Vasi, F., Travisano, M., Lenski, R. E. (1994). Long-term experimental evolution in *Escherichia coli*. II. Changes in life-history traits during adaptation to a seasonal environment. *Am. Natural.* 144(3): 432–456. doi: [10.1086/285685](https://doi.org/10.1086/285685).
- [30] Fox, J. W., Lenski, R. E. (2015). From here to eternity—the theory and practice of a really long experiment. *PLoS Biol.* 13: 1002185.
- [31] Barrick, J. E., Lenski, R. E. (2013). Genome dynamics during experimental evolution. *Nat. Rev. Genet.* 14(12): 827–839. doi: [10.1038/nrg3564](https://doi.org/10.1038/nrg3564).
- [32] Cooper, V. S., Schneider, D., Blot, M., Lenski, R. E. (2001). Mechanisms causing rapid and parallel losses of ribose catabolism in evolving populations of *Escherichia coli* B. *J. Bacteriol.* 183(9): 2834–2841. doi: [10.1128/JB.183.9.2834-2841.2001](https://doi.org/10.1128/JB.183.9.2834-2841.2001).
- [33] Deatherage, D. E., Jamie, K. L., Bennett, A. F., Lenski, R. E., Barrick, J. E. (2017). Specificity of genome evolution in experimental populations of *Escherichia coli* evolved at different temperatures. *Proc. Nat. Acad. Sci. USA* 114: 1904–1912.
- [34] Levy, S. F., Blundell, J. R., Venkataram, S., Petrov, D. A., Fisher, D. S., Sherlock, G. (2015). Quantitative evolutionary dynamics using high-resolution lineage tracking. *Nature*. 519(7542): 181–186.
- [35] Plank, L. D., Harvey, J. D. (1979). Generation time statistics of *Escherichia coli* B measured by synchronous culture techniques. *J. Gen. Microbiol.* 115(1): 69–77. doi: [10.1099/00221287-115-1-69](https://doi.org/10.1099/00221287-115-1-69).
- [36] Rice, S. H. (2004). *Evolutionary Theory*. Sunderland, MA: Sinauer Associates.
- [37] Su, Y., *GreedyGeo (Git Version v0.1-0-g524941d)*, <https://github.com/scopagroup/greedygeo>.
- [38] Schoneman, J. (2016). Stochastic models for genetic evolution, PhD thesis, University of Houston, Houston, TX. <http://hdl.handle.net/10657/3540>.
- [39] Cooper, T. F., Remold, S. K., Lenski, R. E., Schneider, D. (2008). Expression profiles reveal parallel evolution of epistatic interactions involving the CRP regulon in *Escherichia coli*. *PLoS Genet.* 4(2): 35.
- [40] Armijo, L. (1966). Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific J. Math.* 16(1): 1–3. doi: [10.2140/pjm.1966.16.1](https://doi.org/10.2140/pjm.1966.16.1).
- [41] Nocedal, J., Wright, S. J. (2006). *Numerical Optimization*. New York, NY: Springer.
- [42] Boyd, S., Vandenberghe, L. (2004). *Convex Optimization*. Cambridge, UK: Cambridge University Press.