

Article

Estimating the Rate of Mutation to a Mutator Phenotype

Isaac Vázquez-Mendoza ^{1,*} , Erika E. Rodríguez-Torres ¹ , Mojgan Ezadian ² ,
Lindi M. Wahl ²  and Philip J. Gerrish ^{1,3,4,5,*} 

¹ Área Académica de Matemáticas y Física, Universidad Autónoma del Estado de Hidalgo, Pachuca 42039, Hidalgo, Mexico; erikart@uaeh.edu.mx

² Department of Applied Mathematics, Western University, London, ON N6A 3K7, Canada; mezadian@uwo.ca (M.E.); lwahl@uwo.ca (L.M.W.)

³ Biological Sciences, University of Michigan, Ann Arbor, MI 48109, USA

⁴ Department of Biology, University of New Mexico, Albuquerque, NM 87131, USA

⁵ Theoretical Division, Los Alamos National Lab, Los Alamos, NM 87545, USA

* Correspondence: va296000@uaeh.edu.mx (L.V.-M.); pgerrish@unm.edu (P.J.G.)

Abstract: A *mutator* is a variant in a population of organisms whose mutation rate is higher than the average mutation rate in the population. For genetic and population dynamics reasons, *mutators* are produced and survive with much greater frequency than *anti-mutators* (variants with a lower-than-average mutation rate). This strong asymmetry is a consequence of both fundamental genetics and natural selection; it can lead to a ratchet-like increase in the mutation rate. The rate at which mutators appear is, therefore, a parameter that should be of great interest to evolutionary biologists generally; for example, it can influence: (1) the survival duration of a species, especially asexual species (which are known to be short-lived), (2) the evolution of recombination, a process that can ameliorate the deleterious effects of mutator abundance, (3) the rate at which cancer appears, (4) the ability of pathogens to escape immune surveillance in their hosts, (5) the long-term fate of mitochondria, etc. In spite of its great relevance to basic and applied science, the rate of mutation to a mutator phenotype continues to be essentially unknown. The reasons for this gap in our knowledge are largely methodological; in general, a mutator phenotype cannot be observed directly, but must instead be inferred from the numbers of some neutral “marker” mutation that can be observed directly: different mutation-rate variants will produce this marker mutation at different rates. Here, we derive the expected distribution of the numbers of the marker mutants observed, accounting for the fact that some of the mutants will have been produced by a mutator phenotype that itself arose by mutation during the growth of the culture. These developments, together with previous enhancements of the Luria–Delbrück assay (by one of us, dubbed the “Jones protocol”), make possible a novel experimental protocol for estimating the rate of mutation to a mutator phenotype. Simulated experiments using biologically reasonable parameters that employ this protocol show that such experiments in the lab can give us fairly accurate estimates of the rate of mutation to a mutator phenotype. Although our ability to estimate mutation-to-mutator rates from simulated experiments is promising, we view this study as a proof-of-concept study and an important first step towards practical empirical estimation.

Keywords: fluctuation assay; Luria–Delbrück distribution; Jones protocol; mutation rate evolution

MSC: 60J80; 60J85; 92-08; 92-10



Citation: Vázquez-Mendoza, I.; Rodríguez-Torres, E.E.; Ezadian, M.; Wahl, L.M.; Gerrish, P.J. Estimating the Rate of Mutation to a Mutator Phenotype. *Axioms* **2024**, *13*, 117. <https://doi.org/10.3390/axioms13020117>

Academic Editor: Qi Zheng

Received: 10 November 2023

Revised: 24 January 2024

Accepted: 1 February 2024

Published: 11 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Mutation is ultimately the source of new variations that feed natural selection, and the rate at which new mutations appear is, thus, a key parameter for modeling and understanding evolution. To estimate the rate at which a particular mutation appears requires, first of all, that the mutation in question be observable. The famous Luria–Delbrück assay employed a growing bacterial population, which, from a mathematical

standpoint, is a supercritical branching process; our observable mutation appears randomly during this growth process at some low per capita rate, μ , in a way akin to a new “type” appearing at random in a two-type branching process. Our observable mutation is heritable and is assumed to be non-revertible, such that all of its descendants will carry the same observable mutation. In a typical Luria–Delbrück assay, several bacterial cultures are grown in parallel. After a defined period of growth, the numbers of the observable mutation are counted in each culture. Using one of a number of different mathematical models, these mutant counts are used to estimate the per-replication rate μ at which our observable mutation was produced. Much previous work on mutation rates makes an assumption, explicitly or not, that μ is an inherent (and often, by unjustified extrapolation, constant) property of the organism in question. The tables and figures of such mutation rates inherent to different organisms can be found in much-cited previous work [1–4].

For every observable mutation as defined above, i.e., mutations giving rise to mutants that we can observe and count, there will typically be orders of magnitude more mutations that affect some aspect of replication, proofreading, or repair. The number of genes involved in replication, proofreading, and repair can be large and can constitute a significant fraction of an organism’s genome. Any mutation occurring in any of these many genes has the potential to change the mutation rate, and of those that do, most will increase the mutation rate, i.e., will become *mutators*. (This is because random changes in these key genes are more likely to decrease functionality, i.e., to decrease replication fidelity, than to enhance it.)

To give a numerical example, many Luria–Delbrück “fluctuation assays” (including assays conducted by one of us [5]) have employed as their selectable marker (i.e., their observable mutation) resistance to an antibiotic called *Naladixic acid*. Previous studies [6] have revealed that resistance to Naladixic acid, i.e., the ability to observe resistant mutants, is conferred by one or two mutations at the nucleotide level. Given that the *E. coli* genome is roughly five megabases, and given that the wild-type per-nucleotide mutation rate is roughly 0.001, the rate of mutation to observable mutants, in this case, is on the order of 10^{-8} . Given that roughly ten percent (or more) of the *E. coli* genome is devoted to some aspect of replication, proofreading, or repair, the rate of mutation to mutants with potentially different mutation rates (most of which will have increased mutation rates) is on the order of 10^{-4} . The implication is that mutators will appear at a per capita rate that could be up to four orders of magnitude greater than the rate at which the observable mutation appears.

The following question derives naturally from the foregoing observations: How are mutant counts in classical Luria–Delbrück assays (and extensions thereof) affected by the spontaneous appearance of mutator phenotypes during the growth of the cultures? Turning the question around this: If mutant counts are affected by the appearance of mutators during growth, might this fact be leveraged to estimate the rate of mutation to a mutator?

Two previous studies showed remarkable increases in mutator prevalence, hence the mean population mutation rate, owing to the indirect effects of artificial [7] and natural [5] selection. The former study found that a certain mutator phenotype in a specific type of cell would form colonies on plates that looked qualitatively different from non-mutators (or the wild-type). The study was the first of its kind and gave us a ground-breaking first look at the rate of mutation to a mutator phenotype, but it was highly specific. The methodology we present here does not depend on the specific cell type, specific mutator phenotype, or colony morphology. More importantly, the focus of these two previous studies was on the indirect effect of selection on mutator prevalence; while their findings hinted at a significant rate of production of mutators, determining that rate was not their main focus. The present study, on the other hand, seeks to determine the rate at which mutators spontaneously arise in the absence of (or prior to) selection.

In addition to the basic evolutionary questions our experimental protocol can address, there are also significant biomedical applications that can be addressed, not the least of which is cancer research [8–16]. The rate at which mutation rate variants are produced in populations of somatic cells is a key parameter for understanding and predicting somatic

evolution leading to oncogenic transitions; despite its critical role, however, this parameter remains unknown. The work we present here may help to remedy this deficiency.

2. Materials and Methods

2.1. Definitions

A *mutation* is an error that occurs during replication. A *mutant* is an individual that carries a *mutation*. Mutations of interest here are specific mutations (replication errors) that can be observed experimentally. For example, two ways in which specific mutations can be observed are: (1) by eliminating every member in the population that does not carry the specific mutation (for example, applying an antibiotic so that only antibiotic-resistant mutants survive) and (2) by causing a fluorescent marker to fluoresce. Such observable mutations are sometimes called *selectable markers*, and we will refer to such mutations as *marker mutations* and to the rate at which such mutations occur as the *marker mutation rate*. When a mutation occurs, it becomes the founding member of a new *mutant lineage* in which all descendants will carry that *mutation*. In what follows, we use the terms *mutation* and *mutant lineage* interchangeably. For example, we will introduce the concept of a *type- k mutation*, referring to a marker mutation that forms a mutant lineage that ultimately gives rise to k *marker mutants* in the final culture. (We could equivalently say *type- k mutant lineages*.) The protocol we propose is to grow cultures to a final population size that is as large as possible and, then, dilute them (the “Jones protocol” [17–20]), which, as we have shown previously [18], greatly increases the statistical power. In much of what follows, when we mention *final population*, we refer to the population prior to the dilution and plating steps of the Jones protocol. For clarity, Table 1 has been included to present the parameters and notation used in this work.

Table 1. Notation and description of parameters.

N_0, N	initial and final population size.
r	growth rate.
μ	marker mutation rate of wild-type.
μ'	marker mutation rate of mutator.
μ_m	rate of mutation from wild-type to mutator.
m	factor by which wild-type and mutator mutation rates differ: $\mu_m = m\mu$.
$\phi(t)$	recruitment rate of mutations; e.g., $\phi(t) = \mu N_0 e^{rt}$ for marker recruitment rate on the wild-type background.
$p(k; t)$	probability that a mutation appearing at time t leaves k mutants in the final population.
λ_k	expected number of mutations of “type k ”, i.e., that leave k mutants in the final population.
t_f	time at which the population attains its maximum size: $N = N(t_f)$.
Λ	total number of marker mutations that occur on the wild-type background: $\Lambda = \mu(N - N_0) \approx \mu N$.
Λ_m	total number of mutator mutations that occur on the wild-type background: $\Lambda_m = \mu_m(N - N_0) \approx \mu_m N$.
$\varphi(x; \Lambda)$	pgf for the total number of <i>mutants</i> observed in the final population, given that a total of Λ <i>mutations</i> occurred during growth.
$\log$	logarithm base e , i.e., the natural logarithm.
$\log_b$	logarithm base b , for given b .
i	imaginary unit.
$\tilde{f}$	fast Fourier transform of function f .
$\mathcal{L}_i$	likelihood function for the i^{th} growth cycle.

We recall that a mutator is a genotype with some mutation in its replication/proof-reading/repair mechanisms that causes it to replicate with reduced accuracy. Here, we digress briefly to explain the title of our paper: “Estimating the Rate of Mutation to a

Mutator Phenotype". A more-accurate title might be, "Estimating the Rate of Mutation to a Mutation-Rate-Variant Phenotype", as our methods could in theory detect a decrease in the mutation rate, as well as an increase. We have kept the former title, however, because mutator mutations (mutations that increase the mutation rate) are typically loss-of-function mutations and are, therefore, overwhelmingly more common than mutations that decrease the mutation rate (which are gain-of-function mutations).

2.2. Experimental Protocol

1. Grow a bacterial population from initial size N_0 to a large (known) number, N_f , at time t_f .
2. At time t_f :
 - Protocol A: Take two random samples of size N_S (S for the sample of the final population for counting mutants) and N_B (B for the bottleneck sample used to inoculate the subsequent growth cycle), where $N_S \ll N_B \ll N_f$. Use the sample of size N_S to inoculate a number c of flasks with fresh media; these independent cultures grow, and each is then screened for marker mutants. Record the number of marker mutants observed in each of the c cultures. Use the sample of size N_B to inoculate fresh media in a single flask to start the next growth cycle. e.g., see Table 2
 - Protocol B: Take one random sample of size N_B to inoculate fresh medium to start the next growth cycle.
3. Repeat Step 2 $n - 1$ times (for n growth cycles).
4. After the final n^{th} growth cycle, take one random sample of size N_S . Use this sample to inoculate a number c of flasks with fresh media; these independent cultures grow, and each is then screened for marker mutants. Record the number of marker mutants observed in each of the c cultures.
5. See the schematic in Figure 1.

Table 2. Simulated data from Protocol A with three growth cycles. The parameters were $N_0 = 1$, $N = 2^{33}$, $N_B = 2^{20}$, and $N_S = 4295$. This table was included in the hopes that this article, while published in a math journal, might be of interest to both mathematicians and experimentalists. This is an example of what might be recorded in a lab notebook.

Culture	Numbers of Observed Mutants		
	Growth Cycle		
	1	2	3
1	1	6	9
2	1	9	13
3	4	8	19
4	23	8	15
5	5	8	13
6	3	5	14
7	12	3	9
8	1	7	7
9	8	8	13
10	4	5	7
11	1	7	8
12	3	10	12

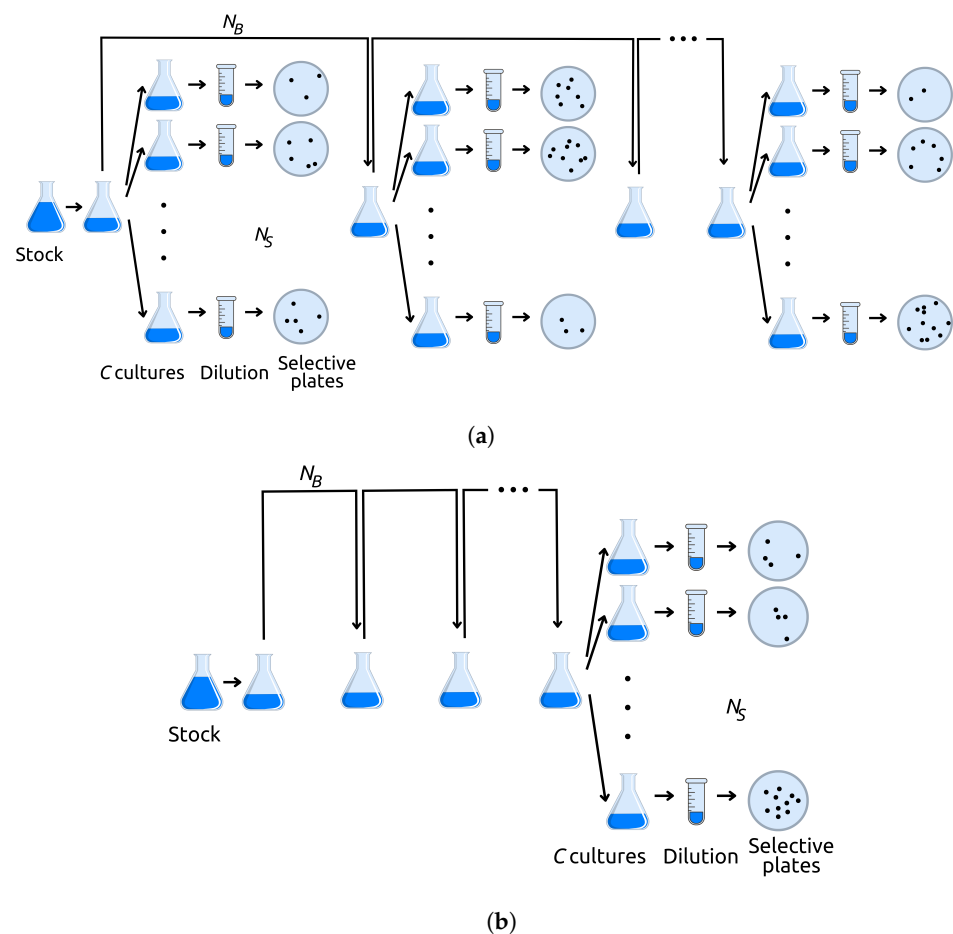


Figure 1. Schematic of the two experimental protocols. To maximize statistical power, the serially passaged populations (i.e., the populations at each growth cycle) should be grown to the largest N_f possible, then diluted to obtain countable numbers of mutants on the selective plates [18]. This takes advantage of the increased statistical power conferred by the “Jones protocol” [18–20], an improvement on the standard fluctuation test [21–25]. (a) Protocol A; (b) Protocol B.

2.3. Analysis

For the simplicity of presentation, we will assume that the population grows exponentially during the growth phase, such that the initial population size is N_0 , and the final population size at time t_f is given by $N = N_0 e^{rt_f}$. We note that our results do not depend on this assumption: we only require that the population grows from size N_0 to N by means of some pure-birth process (meaning that the total number of replications is exactly $N - N_0$). We let $\phi(t)$ denote the “recruitment rate” of mutations at time t , i.e., $\phi(t) = \mu N_0 e^{rt}$, where μ is the mutation rate per individual per unit time. We let $p(k; t)dt$ denote the probability that a mutation appearing in the small time interval $(t, t + dt)$ leaves k mutants in the final population.

We will refer to a mutation (or mutant lineage) as being of “type k ” if it leaves k mutants in the final population. We define random variable M_k to denote the number of mutations (or mutant lineages) of type k . We make the very weak assumption that the M_k are independent, in which case, we have:

$$\mathbb{P}\{M_k = j\} = \frac{\lambda_k^j}{j!} e^{-\lambda_k} \quad (1)$$

where

$$\lambda_k = \int_0^{t_f} p(k; t) \phi(t) dt.$$

The total number of mutants in the final population is:

$$M = \sum_{k=1}^{\infty} k M_k.$$

For a “pure-birth” process of constant birth rate r and assuming that the mutations have no effect on fitness, we know that:

$$p(k; t) = e^{-r(t_f-t)} \left(1 - e^{-r(t_f-t)}\right)^{k-1}$$

(see Ref. [26], p. 450). While we assume r to be a constant here, we note that the results we obtained are robust to this assumption; i.e., any pure-birth regime ($r = r(t) > 0$) should in principle give the same results. This fact may be intuited by recalling that we implicitly made a very weak assumption (i.e., a very reasonable assumption from a biological perspective) that mutation occurs primarily during replication. Under this assumption, what counts is simply the number of replication events and not the dynamics of the replication rates. This assumption can be violated, for example, when ultra-violet radiation is the primary source of mutation, but we suppose that such scenarios are exceptional. From here, we have:

$$\begin{aligned} \lambda_k &= \int_0^{t_f} \mu N_0 e^{rt} p(k; t) dt \\ &= \frac{\mu N}{r} \int_{e^{-rt_f}}^1 u(1-u)^{k-1} du \\ &\approx \frac{\mu N}{k(k+1)}, \end{aligned}$$

given that $r \approx 1$, $e^{-rt_f} = \frac{N_0}{N} \rightarrow 0$, and

$$\int_0^1 u(1-u)^{k-1} du = \frac{1}{k(k+1)}.$$

Now that we derived the overall expected number of mutant lineages of type k , λ_k , and we return to Equation (1) and write the *pgf* for the number of type k mutant lineages as:

$$f_k(x) = e^{\lambda_k(x-1)}.$$

Since each of these lineages contributes exactly k mutants to the final population (prior to dilution), the *pgf* for the total number of mutants contributed by type k lineages to the final population is simply:

$$f_k(x^k) = e^{\lambda_k(x^k-1)}.$$

The *pgf* for the total number of mutants in the final population, from all types, is then given by

$$\varphi(x) = \prod_{k=1}^{\infty} e^{\lambda_k(x^k-1)}.$$

Substituting the expression for λ_k , we find:

$$\varphi(x; \Lambda) = e^{-\Lambda} \exp\left(\sum_{k=1}^{\infty} \frac{\Lambda x^k}{k(k+1)}\right), \quad (2)$$

where $\Lambda = \mu(N - N_0) \approx \mu N$. The notation $\varphi(x; \Lambda)$ emphasizes that the *pgf* for the total number of observed mutants at the final time depends on a single parameter, the product of the final population size and the mutation rate.

We can simplify this expression by considering the following lemma and theorem, whose proofs, despite seeming trivial, constitute an alternative derivation of the Luria–Delbrück *pgf*.

Lemma 1. *Given $z \in [0, 1]$, then*

$$\sum_{j=1}^{\infty} \frac{z^j}{j(j+1)} = \begin{cases} 1 + \left(\frac{1-z}{z}\right) \log(1-z), & \text{if } z \in [0, 1). \\ 1, & \text{if } z = 1. \end{cases}$$

Proof. If $z = 1$, we have that

$$\begin{aligned} \sum_{j=1}^{\infty} \frac{z^j}{j(j+1)} &= \lim_{k \rightarrow \infty} \sum_{j=1}^k \left(\frac{1}{j} - \frac{1}{j+1} \right) \\ &= \lim_{k \rightarrow \infty} \left(\frac{1}{1} - \frac{1}{2} \right) + \left(\frac{1}{2} - \frac{1}{3} \right) + \cdots + \left(\frac{1}{k} - \frac{1}{k+1} \right) \\ &= \lim_{k \rightarrow \infty} 1 - \frac{1}{k+1} \\ &= 1. \end{aligned}$$

For the case $z \in [0, 1)$, let us recall that the Taylor series of $\log(1 + \zeta)$ is

$$T(\zeta) = \sum_{j=1}^{\infty} \frac{(-1)^{j+1}}{j} \zeta^j, \quad \zeta \in (-1, 1),$$

which implies that

$$\begin{aligned} \log(1 - \zeta) &= T(-\zeta) \\ &= \sum_{j=1}^{\infty} \frac{(-1)^{2j+1}}{j} \zeta^j \\ &= - \sum_{j=1}^{\infty} \frac{\zeta^j}{j}, \end{aligned} \tag{3}$$

for each $\zeta \in (-1, 1)$. Therefore,

$$\begin{aligned} \sum_{j=1}^{\infty} \frac{z^j}{j+1} &= \frac{1}{z} \sum_{j=1}^{\infty} \frac{z^{j+1}}{j+1} \\ &= \frac{1}{z} \left(z - z + \sum_{j=2}^{\infty} \frac{z^j}{j} \right) \\ &= \frac{1}{z} \left(-z + \sum_{j=1}^{\infty} \frac{z^j}{j} \right) \\ &= -1 - \frac{1}{z} \log(1 - z). \end{aligned} \tag{4}$$

From Equations (4) and (3), it follows that

$$\begin{aligned}\sum_{j=1}^{\infty} \frac{z^j}{j(j+1)} &= \sum_{j=1}^{\infty} \left(\frac{1}{j} - \frac{1}{j+1} \right) z^j \\ &= \sum_{j=1}^{\infty} \frac{z^j}{j} - \sum_{j=1}^{\infty} \frac{z^j}{j+1} \\ &= -\log(1-z) + \frac{1}{z} \log(1-z) + 1 \\ &= 1 + \left(\frac{1-z}{z} \right) \log(1-z),\end{aligned}$$

as stated. $\square$

Theorem 1. Given $z \in [0, 1]$ and under the convention that $\lim_{(x,y) \rightarrow (0,0)} x^y = 1$, then

$$e^{-\Lambda} \exp \left(\sum_{j=1}^{\infty} \frac{\Lambda z^j}{j(j+1)} \right) = (1-z)^{\frac{\Lambda(1-z)}{z}}. \quad (5)$$

Proof. In the limit as $z \rightarrow 1$, Equation (5) holds by the convention.

Let $z \in [0, 1)$ be a fixed number, and let $\psi = \sum_{j=1}^{\infty} \frac{z^j}{j(j+1)}$. By Lemma 1, we have that

$$\begin{aligned}e^{\Lambda\psi} &= \exp \left(\Lambda + \Lambda \left(\frac{1-z}{z} \right) \log(1-z) \right) \\ &= e^{\Lambda} \exp \left(\Lambda \left(\frac{1-z}{z} \right) \log(1-z) \right) \\ &= e^{\Lambda} \exp \left(\log \left((1-z)^{\Lambda \left(\frac{1-z}{z} \right)} \right) \right) \\ &= e^{\Lambda} (1-z)^{\Lambda \left(\frac{1-z}{z} \right)}.\end{aligned}$$

Thus,

$$e^{-\Lambda} e^{\Lambda\psi} = (1-z)^{\Lambda \left(\frac{1-z}{z} \right)}.$$

$\square$

Applying Theorem 1 to Equation (2), we see that, if a population grows from a single individual to final size N , creating de novo (neutral) mutant lineages at rate μ per individual per unit time, then the total number of mutant individuals in the population at the final time will be described by the pgf:

$$\varphi(z; \Lambda) = (1-z)^{\frac{\Lambda(1-z)}{z}}, \quad (6)$$

which was previously obtained in [27,28]; we will also write:

$$\varphi(z; \Lambda) = h(z)^{\Lambda}, \quad (7)$$

where

$$h(z) = (1-z)^{\frac{1-z}{z}}. \quad (8)$$

Finally, we note that Equation (7) gives the pgf for the number of mutant individuals in a final population of N individuals. Trivially, we re-write as $\varphi(z; \Lambda) = (h(z)^{\mu})^N$; we find that $h(z)^{\mu}$ gives the pgf for the numbers of mutants that a single replication event will leave in the final population. We will use this result in interpreting the expression for $F(z)$ at the end of the following section.

Incorporating Mutators

The foregoing theory does not account for the fact that, as a population grows, variants carrying an elevated mutation rate (mutators) may appear as a consequence of mutations occurring in genes encoding any aspect of replication, proofreading, or repair. We now modify the foregoing theory to account for mutators.

Let $\mu' > \mu$ denote the mutation rate of the mutator to observable mutants (i.e., to the selectable phenotype). Consider a mutator lineage of type k (i.e., a clone that leaves k mutators in the final culture). Since this lineage grows to final size k and produces observable mutants at rate μ' , we can use Equation (6) directly. The number of observable mutants in the final population that were produced within a mutator clone of type k , therefore, has *pgf*:

$$\varphi(z; k\mu') = (1 - z)^{k\mu'(1-z)/z},$$

where this distribution depends again on the product of the final population size—in this case, k —and the mutator mutation rate, μ' . If there are j mutator clones of type k , then the number of observable mutants produced by type- k clones has *pgf*:

$$\varphi(z; k\mu')^j = (1 - z)^{jk\mu'(1-z)/z}$$

The total number of observable mutants produced by type- k mutators has *pgf*:

$$\begin{aligned} \mathbb{E}[\varphi(z; k\mu')] &= \sum_{j=0}^{\infty} \varphi(z; k\mu')^j \mathbb{P}\{M_k = j\} \\ &= \exp\{\lambda_k(\varphi(z; k\mu') - 1)\} \end{aligned}$$

The total number of observable mutants produced by mutators of all types is, therefore, described by the *pgf*:

$$f(z) = \prod_{k=1}^{\infty} \mathbb{E}[\varphi(z; k\mu')]$$

which may be rewritten as:

$$\log f(z) = \sum_{k=1}^{\infty} \lambda_k(\varphi(z; k\mu') - 1) = \sum_{k=1}^{\infty} \lambda_k(h(z)^{k\mu'} - 1)$$

where

$$h(z) = (1 - z)^{(1-z)/z}$$

We define $\Lambda_m = \mu_m N$ to be the recruitment rate of mutators in the population, where μ_m is the rate of mutation to mutators. Substituting the expression for λ_k , using Lemma 1, and letting $g(z) = h(z)^{\mu'}$, we find:

$$\log f(z) = \Lambda_m \sum_{k=1}^{\infty} \frac{g(z)^k - 1}{k(k+1)} = \frac{\Lambda_m(1 - g(z)) \log(1 - g(z))}{g(z)}.$$

The *pgf* for the total number of observable mutants produced by mutators may, thus, be written as:

$$f(z) = (1 - g(z))^{\Lambda_m(1-g(z))/g(z)} = h(h(z)^{\mu'})^{\Lambda_m}.$$

Since the *pgf* describing the total number of observable mutants produced by the wild-type is given by $h(z)^{\Lambda}$ (Equation (6)), the total number observable mutants produced in the culture (i.e., by either wild-type or mutator), therefore, has *pgf*:

$$F(z) = h(z)^{\Lambda} h(h(z)^{\mu'})^{\Lambda_m} \quad (9)$$

This expression has a clear intuitive explanation. We know that $h(\cdot)^{\Lambda_m}$ describes the total number of mutators produced by the wild-type in the final population. Since the

pgf for the number of observable mutants produced by a single replication event within a mutator lineage is given by $h(z)^{\mu'}$, the composition $h(h(z)^{\mu'})^{\wedge m}$ gives the pgf associated with the total number of observable mutants in the final population produced by mutators. The product of this term with the pgf for the observable mutants produced by the wild-type, $h(z)^{\wedge}$, then gives the pgf for the total number of observable mutants in the final population.

2.4. Dilution Step

We have previously shown [18] that a slight modification of an experimental protocol initially proposed by Jones [19,20] can significantly increase the accuracy with which mutation rates are estimated and the statistical power to distinguish between different mutation rates. The protocol requires that cultures be grown to as large a size as possible; the numbers of observable mutants in the final populations will be very large and will require a dilution step to obtain a number of mutants that can be counted on the selective plates (see Figure 1). To account for the dilution step, we simply compose $pgf F(z)$ with the pgf for Bernoulli sampling:

$$\sigma(z) = (1 - p) + pz,$$

where p is the fraction of the final population to be plated (i.e., it is the inverse of the dilution factor). The composition $F(\sigma(z))$ is, thus, the pgf for numbers of mutants observed on plates.

2.5. Simulated Data

Our overall goal is to use the analytical results described above to estimate the underlying mutation rates in populations that have been propagated as described in Section 2.2. To test this idea, we generated simulated data using three independent approaches.

In the pure-birth process simulation, we simulated a pure-birth process for the population growth. A random individual in the population was chosen to replicate and was replaced by two daughter cells. If the parent individual is a wild-type cell, each daughter can independently mutate to the marker or mutator phenotype. If the parent carries the marker mutation, daughters can further mutate to the marker–mutator phenotype, while if the parent is a mutator, the daughters have a higher probability of mutating and can likewise mutate to the marker–mutator phenotype. The birth process is repeated until the population reaches a specified final population size. The pure-birth process simulation is computationally expensive, but offers a direct simulation of the stochastic process described by the analytical approach. See Appendix A.3 for a detailed description and the pseudocode.

In the graphical model, the occurrence of possible mutations in a growing population was simulated by generating uniformly distributed random points on the rectangle $[0, t_f] \times [0, 1]$. The expected growth curve of the wild-type population was then computed and normalized by the final population size, N . This normalized growth curve was used as a discriminant function; in particular, points in the rectangle described above that fall below the growth curve indicate mutations in the wild-type strain, as shown in Figure A1. This process was repeated for both types of possible mutations to the wild-type (marker and mutator mutations). The resulting mutant lineages were then, in turn, subject to the same procedure to identify mutations to lineages carrying both marker and mutator mutations.

The graphical model, named after the graphical selection of mutations described in Figure A1, was implemented because it is relatively straightforward to understand and computationally efficient at biologically realistic mutation rates. This model has the disadvantage, however, of generating redundant information since cells of all types, in particular mutators that do not display the marker phenotype, are tracked, despite our interest in cells carrying the marker phenotype only. For a detailed description, including the pseudocode, see Appendix A.1.

As an alternative method, we employed the standard inverse CDF method—which we will call the “quantile method”—for generating random numbers from a given probability distribution, in which a random number is generated in the interval $[0, 1]$ and is mapped

onto the CDF to generate the random number. In this case, the CDF employed was simply the normalized growth curve of mutant cells (i.e., normalized so that the minimum is 0 and the maximum is 1). In other words, we employed the growth curve as if it were the CDF. Since the probability of a mutation appearing in some individual is directly proportional to the number of individuals present, this methodological extrapolation is entirely natural. This procedure was employed in a nested manner in order to simulate not only the appearance of mutator lineages, but also to simulate the appearance of observable mutant lineages within the mutator lineages.

Unlike the previous two methods, the computational execution of the quantile method does not produce redundant information, making it the fastest alternative among the presented computational methods. However, since the mathematical underpinnings of this method are not as intuitively clear as for the other two approaches, we implemented all three approaches for validation.

For a detailed description of the quantile method, see Appendix A.2. Both the graphical and quantile methods were developed in an R package available at <https://github.com/isaacvazquez1/EstimatingMutationRates>, accessed on 31 January 2024.

Before proceeding with the parameter estimation, we first tested the validity of these three approaches in creating simulated data. Using the simple case of a single bottleneck and no mutators as a test case, we compared that datasets produced by the three approaches to each other and to the Luria–Delbrück distribution, as described in detail in Section 3.1.

2.6. Parameter Estimation

After validating the three simulation approaches, we used the simulated data to test whether our analytical model can be used to estimate the underlying mutation rates, in particular the mutation rate to the mutator phenotype. To do this, we developed three methods to estimate the mutation-to-mutator rate from the simulated data.

In the direct estimation method, the distance between the empirical probability-generating function defined in Equation (A25) and its theoretical counterpart displayed in Equation (9) is minimized by applying the least-squares method, finding the best estimate for the rate of mutation μ_m . In particular, we directly minimized the mean square error of the *pgf* to provide estimates of $\Lambda = N\mu$, $\Lambda_m = N\mu_m$, and μ' . Assuming N is known, this yields the desired estimate of μ_m .

The power of this method relies on the simplicity of the ideas underlying the widely known least-squares methodology. Nevertheless, bootstrapping was employed to build the $(1 - \alpha)100\%$ confidence interval, a statistical technique that could fail when heavy-tailed distributions are concerned [29]. This approach to directly estimating the underlying mutation rates is described in detail in Appendix B.1.

In addition to direct *pgf*-based parameter estimation, we developed two related approaches to parameter estimation based on maximum likelihood estimation (MLE).

In the MLE approach, an approximation of the probability mass function (*pmf*) of the random variable whose *pgf* is given in Equation (9) was obtained by using the fast Fourier transform (FFT), as presented in Equation (A29). This method then uses maximum likelihood methods to fit the empirical *pmf* to the approximated *pmf*, so that the maximum likelihood estimators of Λ , Λ_m , and μ' can be computed.

This method was included because, in many contexts, MLE methods are known to be more accurate (and sometimes, more computationally efficient) than *pgf*-based estimation methods (but, please see [24] and the references therein, which may lead to updated and improved methods here).

In our approaches, it was assumed that there are only two possible mutation rates, namely μ and $\mu' = k\mu$, where the parameter $k \geq 0$ is the “mutator strength” and is assumed to be constant and known.

The above simplification aids in streamlining the computational execution; however, any error in the estimate of k may induce bias in the estimated parameter value. Ideally, we would like to be able to leave k as an additional parameter to be estimated. In theory, this is

possible, but given the amount of data typically acquired from the lab, an ability to further estimate k seems dubious. Indeed, if the lab produced enough data, it would in theory be possible to estimate a distribution of the random variable K representing mutator strength. These considerations serve to illustrate the constraints on the methodologies we present owing to experimental limitations.

Both of the MLE approaches are described in detail in Appendix B.2. All of our parameter estimation methods were developed in an R package available at <https://github.com/isaacvazquez1/EstimatingMutationRates>, accessed on 31 January 2024.

3. Results

3.1. Model Reliability

We first validated the reliability of our data simulation approaches, both by comparing them against each other and by comparing the simulated data to the known Luria–Delbrück distribution.

A cell population where only observable (marker) mutations are allowed, such that no mutators or marker cells with mutator phenotypes appear, can be simulated in the pure-birth simulation by setting $\mu_m = 0$ and can also be modeled with the graphical model (described in Appendix A.1) and the quantile function model (described in Appendix A.2), stopping them at Equations (A5) and (A18), respectively.

For each of the final population sizes $N \in \{20,000, 40,000, 60,000, 80,000, 100,000, 200,000, 300,000, 400,000, 500,000, 600,000, 700,000, 800,000, 900,000, 1,000,000\}$, 100 simulations were generated by the pure-birth, graphical, and the quantile function models, respectively. Goodness-of-fit tests were carried out to determine whether the null hypothesis, H_0 : the datasets are drawn from the same distribution, can be rejected. Pairwise two-sample Kolmogorov–Smirnov tests for discrete distributions [30] were performed, as used in [25].

As shown in Figure 2, the distributions associated with the simulations generated by pure-birth, graphical, and quantile function models were not significantly different ($p > 0.05$) for all final population sizes tested.

Moreover, histograms of the number of mutant cells reported in the datasets built by the quantile, graphical, and pure-birth models were examined for a visual goodness-of-fit analysis. To illustrate the case when the p -value is approximately one for all three comparisons, as shown in Figure 2a–c, $N = 700,000$ was chosen; in this case, the smallest p -value attained was 0.9062064 (see Figure 2b). The comparison of the distributions is displayed in Figure 2d. In contrast, to present the worst-fit case in any of the three comparisons, $N = 800,000$ was chosen since the smallest p -value obtained was 0.28096 (see Figure 2b); the comparison of the distributions can be found in Figure 2e.

Furthermore, the distribution underlying the datasets is known since the number of mutants produced by wild-type cells has the Luria–Delbrück distribution [27] (Equation (15)). For each dataset with a final population size $N = 1,000,000$, used for Figure 2, analysis using the web tool *bz-rates* [31] was performed.

The *bz-rates* platform provides visual verification of the fitting of datasets to the Luria–Delbrück distribution, an estimation of the mutation rate, and the 95% confidence interval for the mutation rate. In addition, a χ^2 goodness-of-fit test was performed to determine whether the null hypothesis, H_0 : the dataset was drawn from a Luria–Delbrück distribution, can be rejected.

As shown in Figure 3, it was not possible to reject the null hypothesis that the datasets were drawn from a Luria–Delbrück distribution, with the Luria–Delbrück and dataset distributions not being statistically different ($p > 0.05$); see Table 3.

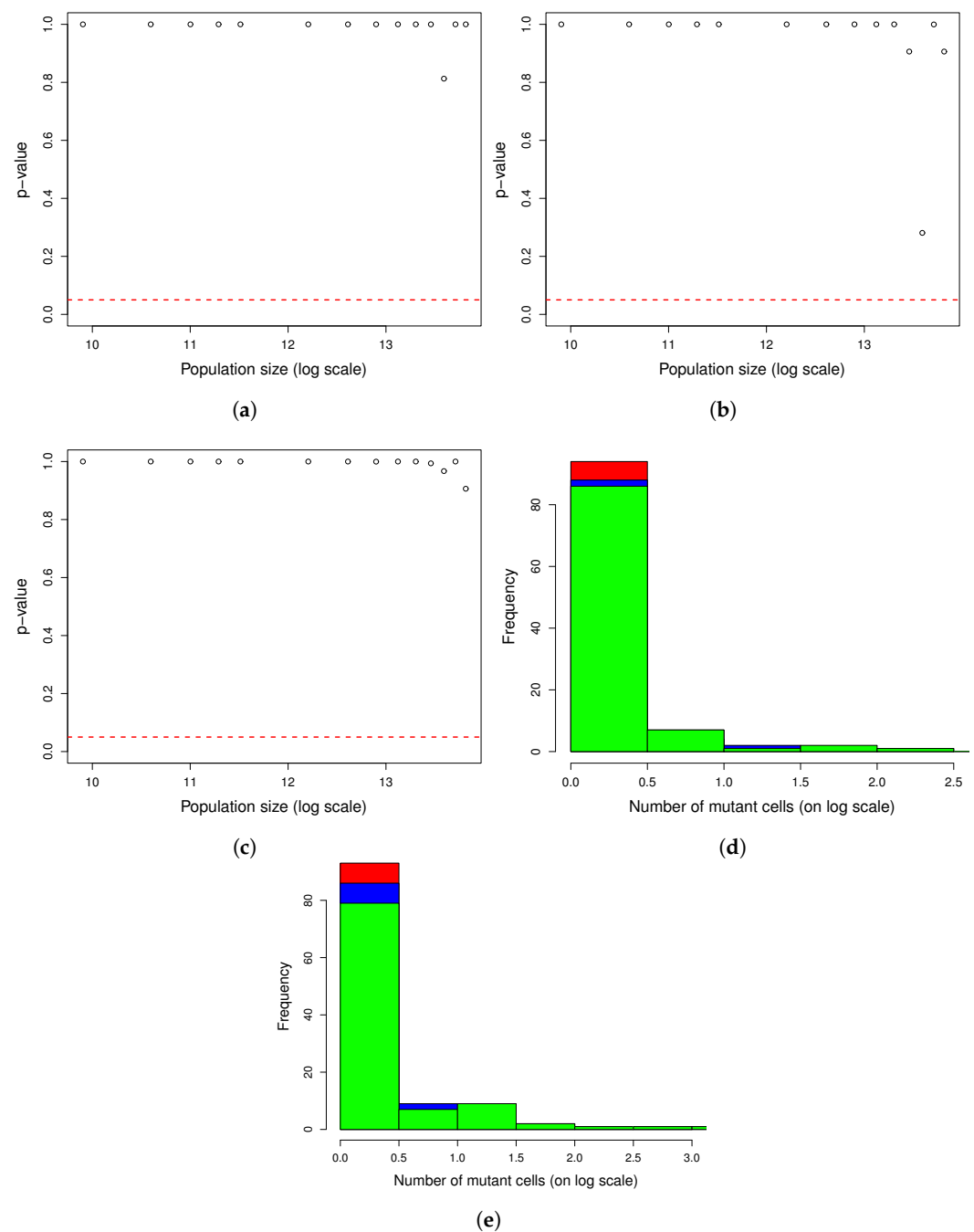


Figure 2. Statistical comparison of dataset distributions. Given a final population size N , $N \in \{20,000, 40,000, 60,000, 80,000, 100,000, 200,000, 300,000, 400,000, 500,000, 600,000, 700,000, 800,000, 900,000, 1,000,000\}$, 100 simulations were generated with each of the models. For the three models, the parameters were $\mu = 10^{-7}$, $\mu' = 0$, and $\mu_m = 0$; hence, only mutant cells produced by wild-type cells were allowed. The horizontal red dashed line shows the upper bound of the rejection region ($p = 0.05$) for the null hypothesis H_0 , which states that both datasets come from the same distribution. A Kolmogorov–Smirnov test was performed per final population size, reporting the p -values, plotted as open circles, associated with (a) distribution comparisons between the datasets of the pure-birth and graphical models, (b) distribution comparisons between the datasets of the pure-birth and quantile function models, and (c) distribution comparisons between the datasets of the graphical and quantile function models. Additionally, given a final population size, histograms of the number of mutant cells reported in the datasets of the quantile, graphical, and pure-birth models are displayed in red, blue, and green, respectively. Here, (d) shows the histograms corresponding to $N = 700,000$ and (e) shows the histograms corresponding to $N = 800,000$.

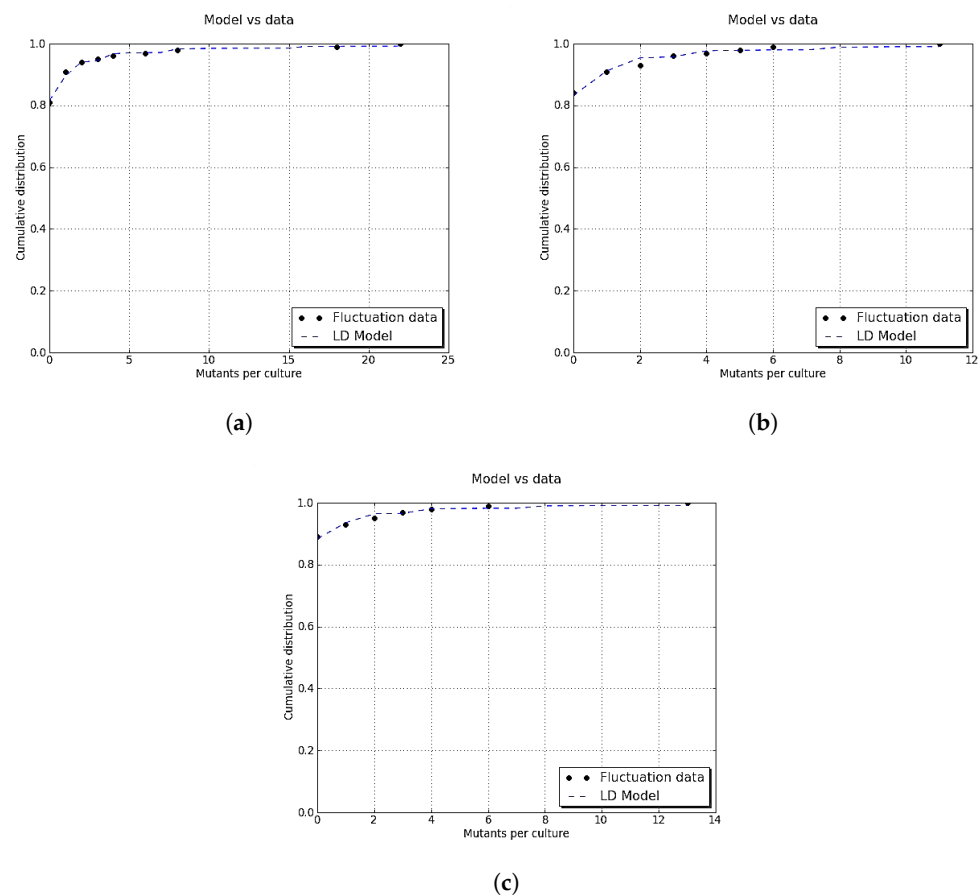


Figure 3. Fitting datasets of mutants produced by wild-type cells to the Luria–Delbrück distribution. Datasets for final population size $N = 1,000,000$, used for Figure 2, were used to visually verify their fit to the Luria–Delbrück distribution. The blue dashed line shows the cumulative distribution function associated with the Luria–Delbrück distribution, and the black dots show the empirical cumulative distribution function associated with the dataset. (a) Fitting of the pure-birth process simulations. (b) Fitting of the graphical model simulations. (c) Fitting of the quantile function model simulations.

Table 3. Parameter estimation analysis on *bz-rates*. Datasets of mutants produced by wild-type cells in a final population size $N = 1,000,000$ with mutation rates $\mu = 10^{-7}$, $\mu' = 0$, and $\mu_m = 0$, used for Figure 2 were taken.

	Pure-Birth	Graphical	Quantile Function
Estimate of μ	2.05063×10^{-7}	1.841444×10^{-7}	1.237306×10^{-7}
Estimation error	1.05063×10^{-7}	8.4144×10^{-8}	2.3730×10^{-8}
95% confidence interval lower bound	1.262×10^{-7}	1.098×10^{-7}	6.346×10^{-8}
95% confidence interval upper bound	2.839×10^{-7}	2.585×10^{-7}	1.840×10^{-7}
Pearson's χ^2 value	1.6333	4.1370	1.0446
<i>p</i> -value	0.8027	0.3877	0.9029

3.2. Reliability of Parameter Estimation

By inspection, it was noted that considering more than one growth cycle provided no estimation improvement. Thus, we present the estimates and confidence intervals for the first growth cycle only, obtained by three methods, described in detail in Section 3.2: (1) bootstrap confidence interval and point estimation [32] (Chapter 13) applied to the direct estimation of μ_m (direct estimations bootstrap), (2) the likelihood-based confidence region [33] (Equation (14)) [34] applied to the maximum likelihood estimation of μ_m (MLE),

and (3) the likelihood-based confidence region, where the mutator strength was fixed, applied to the maximum likelihood estimation of μ_m (two-variable MLE). See Figures 4–7.

To standardize the estimation methods, given a mutation rate $\eta \in \{\mu, \mu', \mu_m\}$, we define the interval $J \in \{I, I', I_m\}$ as

$$J = [\log_{10}(\eta) - 2, \log_{10}(\eta) + 2]. \quad (10)$$

Moreover, parameter p^* was set as 20 and since only the first growth cycle will be considered $c^* = 1$.

Three datasets were created to test the estimation methods, with $r^* = 125$ replicates for the dataset built with the pure-birth simulation and $r^* = 150$ replicates for the datasets built with the graphical and quantile function models. The first dataset was built with the pure-birth simulation (parameters: $N = 2^{33}$, $t_f = 33$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-5}$, $\mu_m = 10^{-4}$); the second one was built with the quantile function model (parameters: $N = 2^{33}$, $t_f = 33$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-4}$, $\mu_m = 10^{-5}$), and the last one was built with the graphical model (parameters: $N = 2^{33}$, $t_f = 33$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-5}$, $\mu_m = 10^{-6}$).

For the direct estimations bootstrap method, a total of 500 estimations were performed, maintaining the dilution parameters as $\delta_B = 2^{20}/2^{33} = 0.0001220703$ for dilution due to bottlenecks between growth cycles and $\delta_P = 0.00001$ for dilution used for plating and counting mutants.

For the MLE method, a single estimation was performed setting $\delta_B = 2^{20}/2^{33}$.

Lastly, for the two-variable MLE, a single estimation was performed setting $\delta_B = 2^{20}/2^{33}$, and μ' was taken as $k\mu$, where $k > 1$ denotes that the mutator strength was chosen to be $k = 500$ when the datasets built with the pure-birth simulation and graphical method were used to perform the estimations and $k = 5000$ when the datasets built with the quantile method were used to perform the estimations.

Given a dataset, a performance comparison between the three methods was conducted. For comparisons related to the dataset built with the pure-birth simulation, the quantile function method, and the graphical method see Tables 4–6, respectively.

Table 4. Estimates and 95% confidence intervals for a dataset built with the pure-birth simulation. For the dataset, the following parameters were considered: $N = 2^{33}$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-5}$, and $\mu_m = 10^{-4}$.

Method	Estimate of μ_m	95% Confidence Interval
Direct estimations	1.25197×10^{-4}	$(1 \times 10^{-6}, 8.575959 \times 10^{-4})$
Bootstrap		
MLE	5.858058×10^{-3}	$(1 \times 10^{-6}, 1 \times 10^{-2})$
Two-variable		
MLE	8.157906×10^{-6}	$(1 \times 10^{-6}, 3.434776 \times 10^{-4})$

Table 5. Estimates and 95% confidence intervals for a dataset built with the quantile function model. For the dataset, the following parameters were considered: $N = 2^{33}$, $t_f = 33$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-4}$, and $\mu_m = 10^{-5}$.

Method	Estimate of μ_m	95% Confidence Interval
Direct estimations	1.958892×10^{-4}	$(1 \times 10^{-7}, 1 \times 10^{-3})$
bootstrap		
MLE	3.704046×10^{-7}	$(1 \times 10^{-7}, 7.416085 \times 10^{-4})$
Two-variable		
MLE	2.604188×10^{-6}	$(1 \times 10^{-7}, 1.416185 \times 10^{-5})$

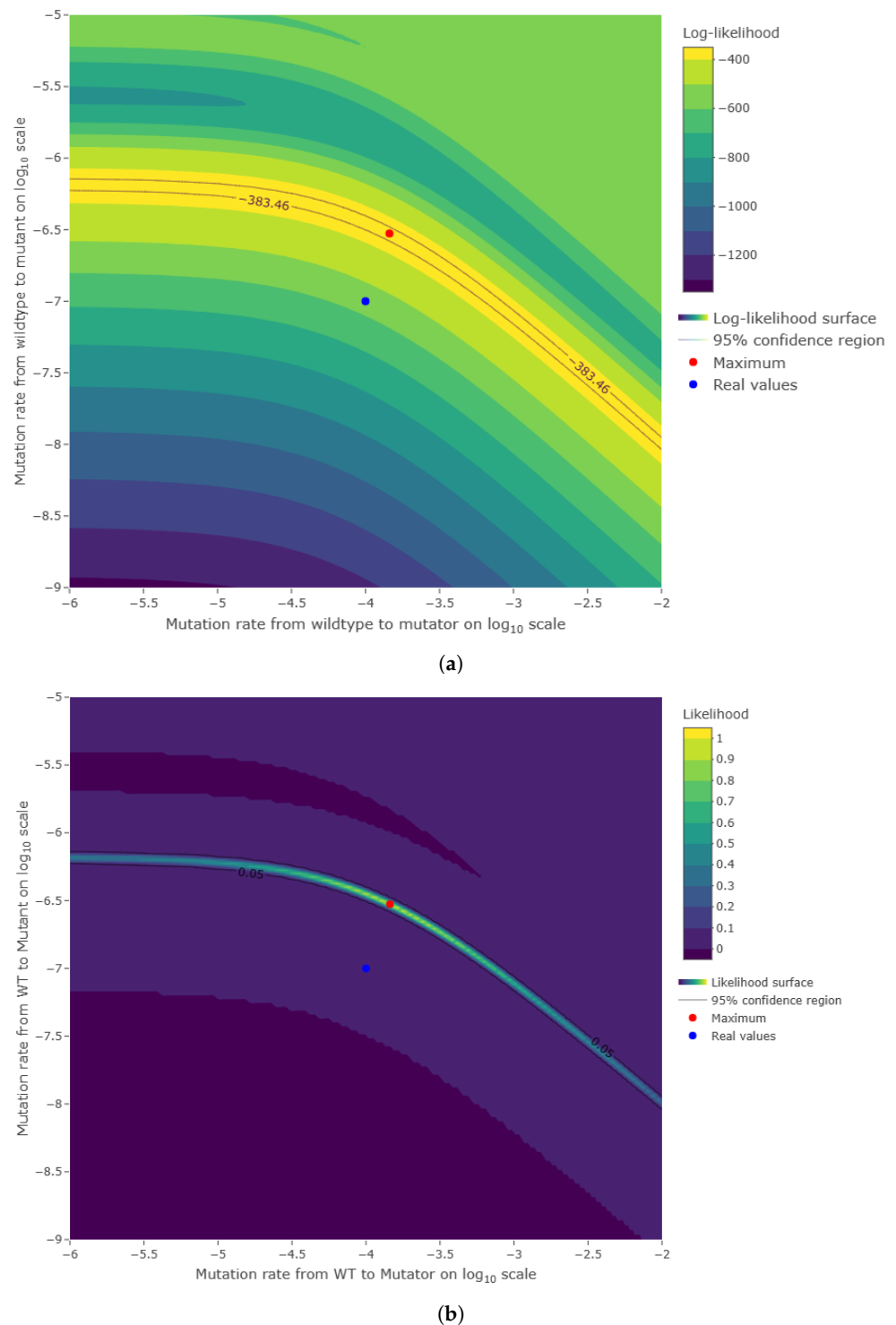


Figure 4. Contour graph of the log-likelihood function and likelihood function, respectively. Given the dataset built with the pure-birth simulation used for Table 4, (a) shows the log-likelihood function $\log(\mathcal{L}_1)$, where the mutator strength was fixed (two-variable MLE), was constructed, while (b) shows the likelihood function multiplied by the scaling factor $\left(\max_{\omega \in \Omega} \{\mathcal{L}_1(\omega)\}\right)^{-1}$. All parameters remain unchanged except for $p^* = 150$, instead of $p^* = 20$, for better image resolution. The red dot shows where the maximum of the function was attained; the blue point displays the actual mutation rates; the region between the black curves represents the 95% confidence region.

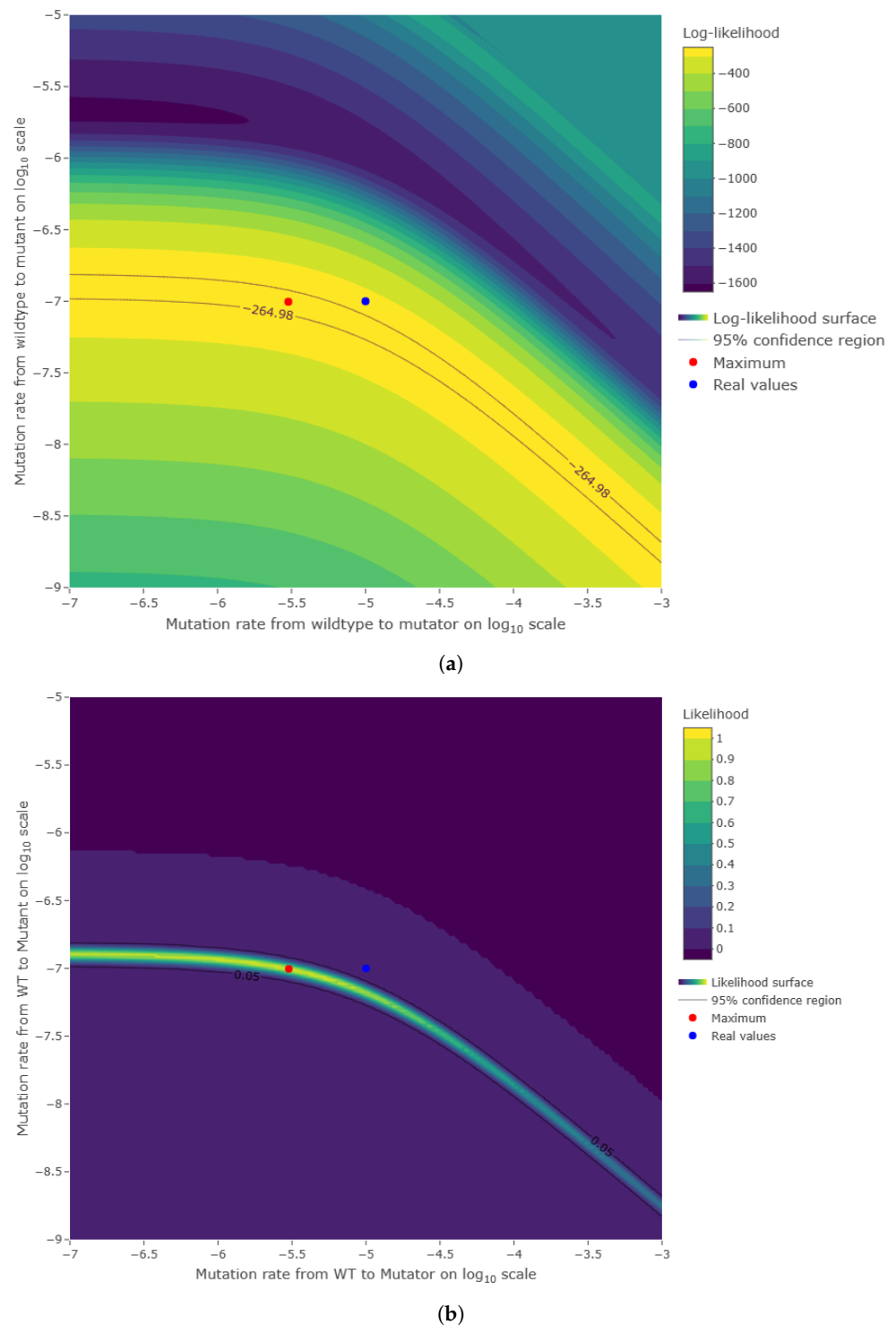
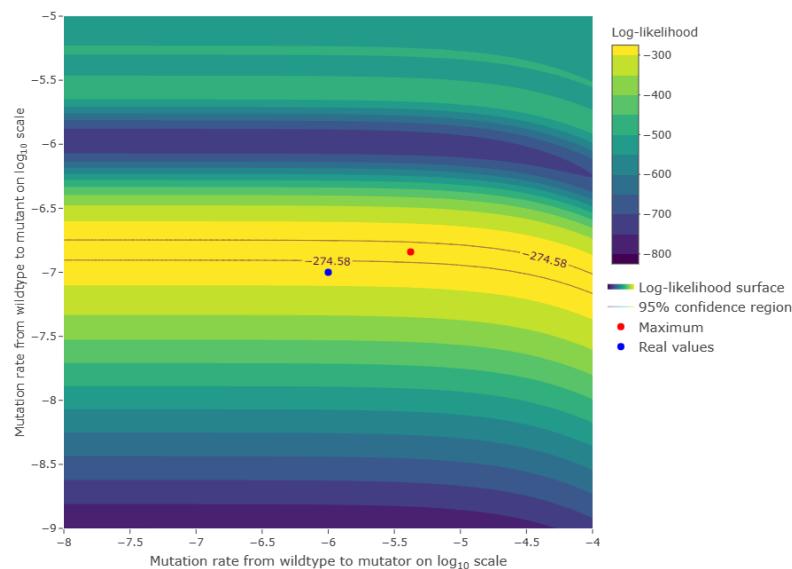


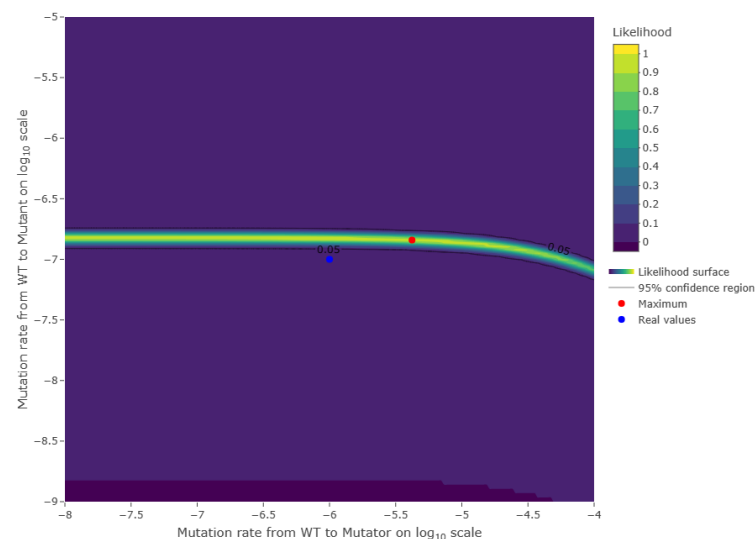
Figure 5. Contour graph of the log-likelihood function and likelihood function, respectively. Given the dataset built with the quantile method used for Table 5, (a) shows the log-likelihood function $\log(\mathcal{L}_1)$, where the mutator strength was fixed (two-variable MLE), was constructed, while (b) shows the likelihood function multiplied by the scaling factor $\left(\max_{\omega \in \Omega} \{\mathcal{L}_1(\omega)\}\right)^{-1}$. All parameters remain unchanged except for $p^* = 150$, instead of $p^* = 20$, for better image resolution. The red dot shows where the maximum of the function was attained; the blue dot displays the actual mutation rates; the region between the black curves represents the 95% confidence region.

Table 6. Estimates and 95% confidence intervals for a dataset built with the graphical model. For the dataset, the following parameters were considered: $N = 2^{33}$, $t_f = 33$, $\mu = 10^{-7}$, $\mu' = 5 \times 10^{-5}$, and $\mu_m = 1 \times 10^{-6}$.

Method	Estimate of μ_m	95% Confidence Interval
Direct estimations bootstrap	4.446193×10^{-6}	$(1 \times 10^{-8}, 1.165914 \times 10^{-5})$
MLE	8.652343×10^{-7}	$(1.874255 \times 10^{-7}, 1 \times 10^{-4})$
Two-variable MLE	1.656201×10^{-5}	$(2.168967 \times 10^{-6}, 9.211989 \times 10^{-5})$



(a)



(b)

Figure 6. Contour graph of the log-likelihood function and likelihood function, respectively. Given the dataset built with the graphical method used for Table 6, (a) shows the log-likelihood function $\log(\mathcal{L}_1)$, where the mutator strength was fixed (two-variable MLE), was constructed, while (b) shows the likelihood function multiplied by the scaling factor $\left(\max_{\omega \in \Omega} \{\mathcal{L}_1(\omega)\}\right)^{-1}$. All parameters remain unchanged except for $p^* = 150$, instead of $p^* = 20$, for better image resolution. The red dot shows where the maximum of the function was attained; the blue dot displays the actual mutation rates; the region between the black curves represents the 95% confidence region.

Additionally, the consistency of the estimation methods presented herein was explored by taking the datasets used for the tables above to perform each method, as previously described, a total of 100 times. Replicates of estimations for the datasets used for Tables 4–6 are shown in Figures 8–10, respectively.

Table 6 led us to conjecture that the estimates obtained using the data generated with the graphical model were the most consistent. Therefore, in Figure 7, the behavior of the estimates of μ_m and the confidence intervals was studied; all the parameters were kept except for μ_m , which varied, taking values of $\{10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}\}$.

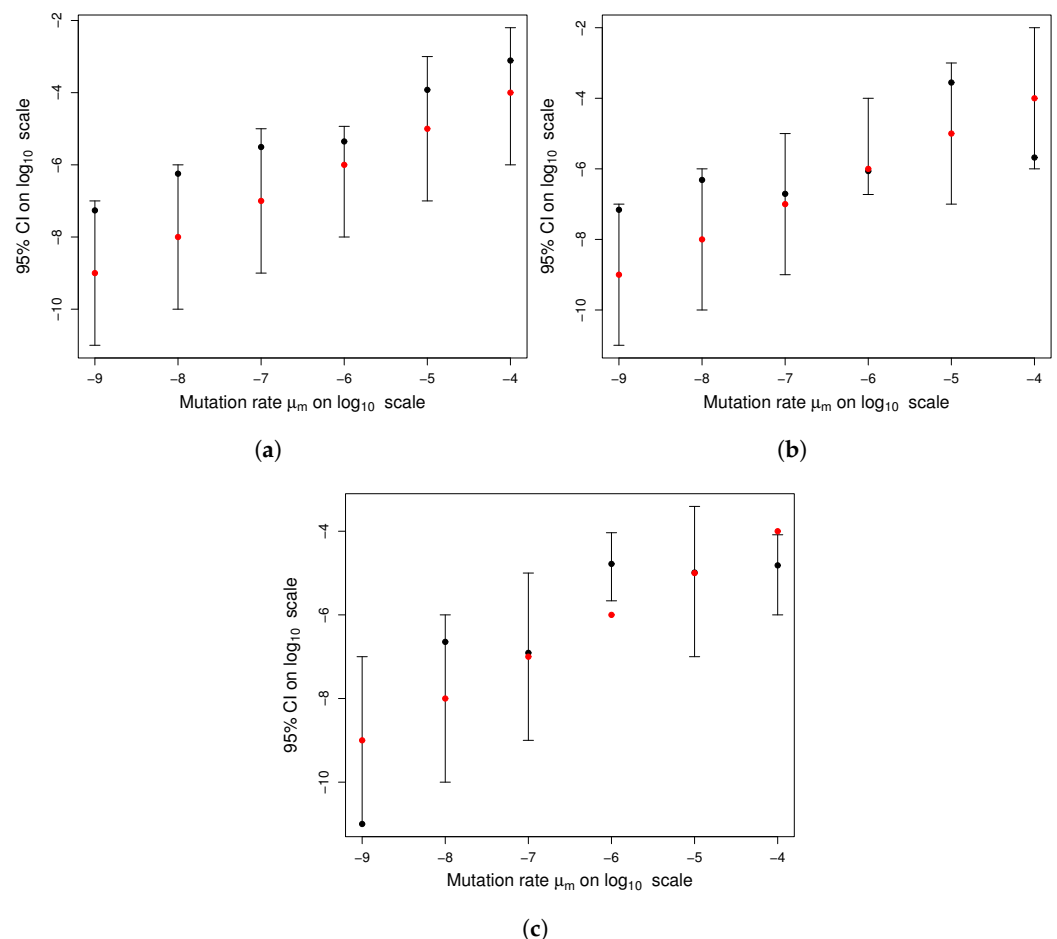


Figure 7. Estimates of μ_m and the corresponding 95% confidence interval for datasets built with the graphical model with μ_m varying. All model and estimation method parameters are as used in Table 6, while $\mu_m = 10^q$, where $q \in \{-9, -8, -7, -6, -5, -4\}$. Each estimation method was applied a single time per dataset using (a) the direct estimations bootstrap, (b) the MLE, and (c) the two-variable MLE methods, respectively. The red dot shows the real value of μ_m ; the mutation rate estimations are displayed as black dots, while the respective 95% confidence interval is shown vertically.

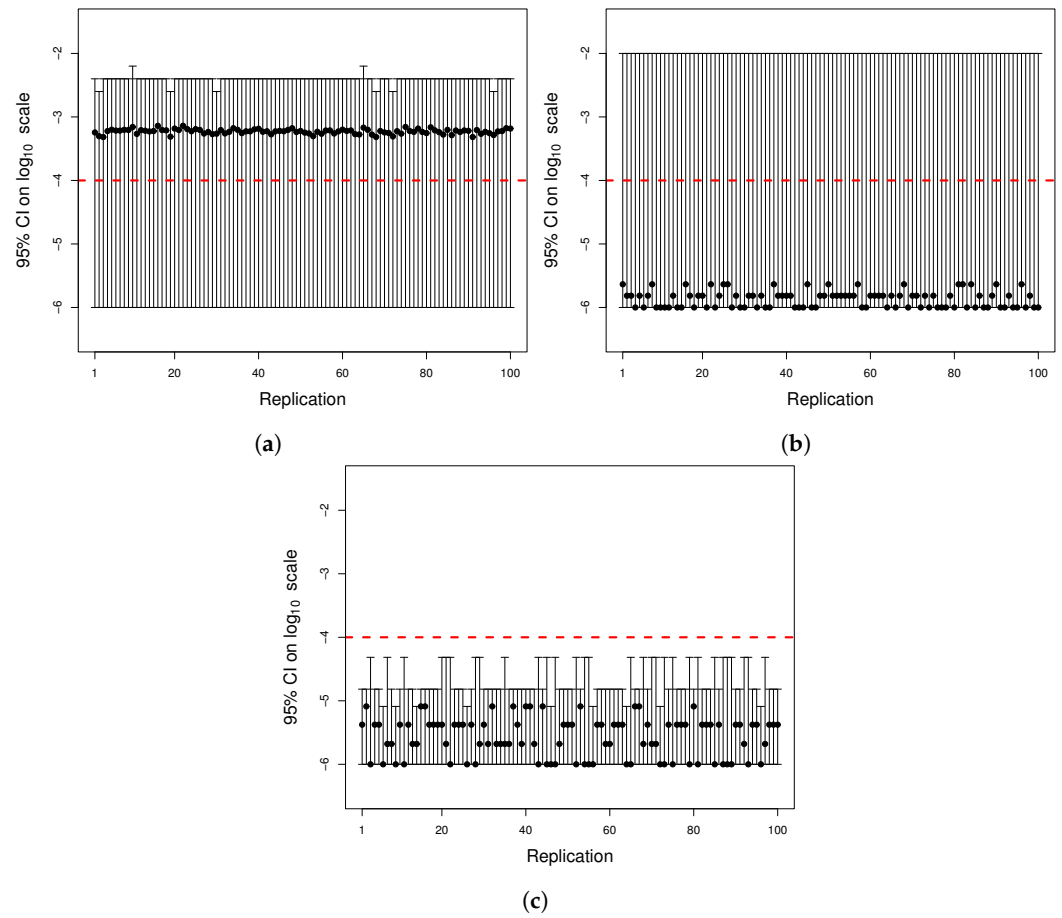


Figure 8. Replications of the 95% confidence interval for the dataset built with the pure-birth simulation used in Table 4. A total of 100 replications were performed per estimation method, (a) replicates using the direct estimation method, (b) replicates using the MLE method, and (c) replicates using the two-variable MLE method. The red dashed line shows the real value of the mutation rate ($\mu_m = 10^{-4}$); the mutation rate estimations are displayed as black dots, while their respective 95% confidence interval is shown vertically.

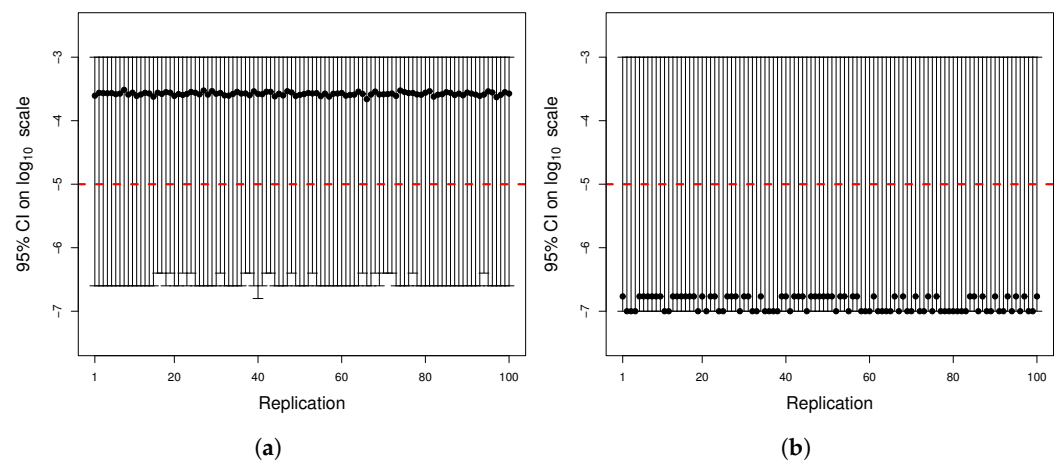
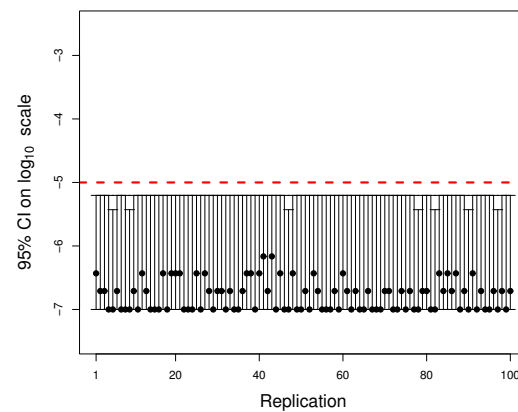
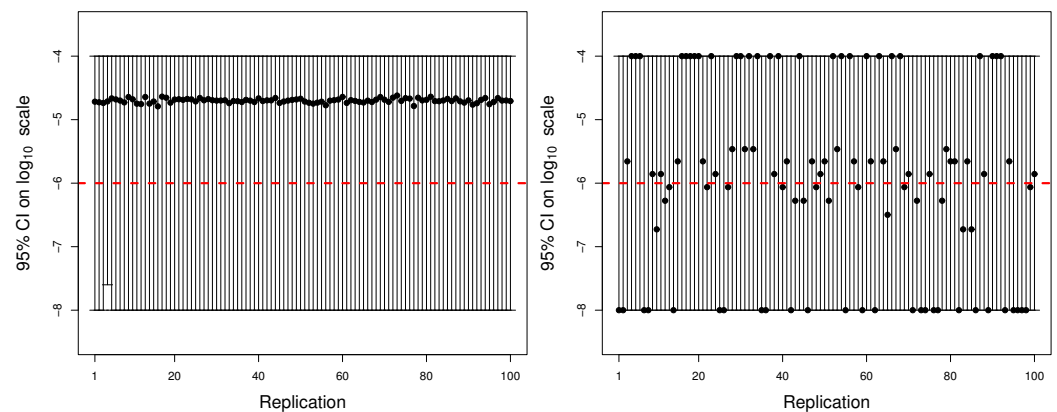


Figure 9. Cont.



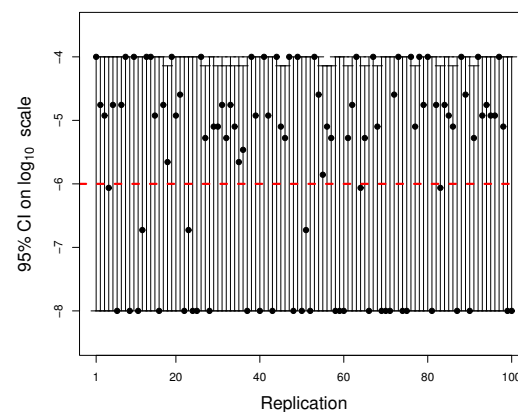
(c)

Figure 9. Replications of the 95% confidence interval for the dataset built with the quantile model used in Table 5. A total of 100 replications were performed per estimation method, (a) replicates using the direct estimation method, (b) replicates using the MLE method, and (c) replicates using the two-variable MLE method. The red dashed line shows the real value of the mutation rate ($\mu_m = 10^{-6}$); the mutation rate estimations are displayed as black dots, while their respective 95% confidence interval is shown vertically.



(a)

(b)



(c)

Figure 10. Replications of the 95% confidence interval for the dataset built with the graphical model used in Table 6. A total of 100 replications were performed per estimation method, (a) replicates using the direct estimation method, (b) replicates using the MLE method, and (c) replicates using the two-variable MLE method. The red dashed line shows the real value of the mutation rate ($\mu_m = 10^{-6}$); the mutation rate estimations are displayed as black dots, while their respective 95% confidence interval is shown vertically.

4. Discussion

Our results suggest that a feasible experimental protocol—a variant of the famous Luria–Delbrück experiment—for estimating the rate of mutation to mutator phenotypes may not be out of reach. To assess this possibility, we explored a parameter space within which we know the relevant parameters lie. There is, however, one parameter that presents us with a potential caveat; somewhat vexingly, this parameter is precisely the one that we are most interested in, namely the rate of mutation to mutator. The reason for this caveat lies in the fact that we pointed out at the very beginning of this manuscript: aside from hand-wavy gene-counting arguments, we have little *a priori* knowledge of what a range of plausible values might be for the rate of mutation to mutator. This is of course not a problem for assessing our methods by comparison with simulation. It is a problem, however, when assessing the practical feasibility of our proposed protocol in the lab: if the rate of mutation-to-mutator is in fact extremely low (very doubtful, but possible), the population sizes required for its estimation may become experimentally unfeasible.

We proposed and analyzed different numerical estimation methods for the estimation of three different mutation rates, of which primary interest is centered on the mutation-to-mutator phenotype, as the title of this article implies. The different methods have different pros and cons, but estimation accuracy is similar across the different methods. Further studies are required to optimize the methods and perhaps reach a verdict on which of the estimation methods provides the most-accurate estimates and the greatest statistical power. Our MLE methods can be improved upon, which directly fit the *pgf*'s to their empirical counterparts. We have not yet explored some of the refinements of the later developed by Ycart and others [24,25,35–38].

In general, the estimations provided by our methods are encouraging, yet wanting. Regarding the direct *pgf* estimation methods, they tend to overestimate μ_m and require a considerable amount of computational time. With respect to the MLE and the two-variable MLE methods, they provide estimates of μ_m faster with comparable accuracy.

One parameter whose effect we have not explored is the bottleneck size: we only employed a bottleneck size of 2^{20} . With this bottleneck size, we found that applying more than one growth cycle did not yield much improvement in the estimates. It may be that for a different bottleneck size, estimates can be improved by performing more than one growth cycle.

The present manuscript has provided a springboard from which we hope ourselves and others will “take the baton”, so to speak, and run with these ideas towards something that is informative (and concretely useful to applied scientists working in infectious diseases, oncology, etc.) under experimentally reasonable conditions. The challenges that lie ahead are both experimental and theoretical. For example, how might we leverage the concepts of MCMC and importance sampling to improve the accuracy of our estimates and/or require fewer experimental data? In a context of somatic cell growth and evolution, how might detailed-balance and Metropolis–Hastings sampling improve our forecasting and analysis of the onset of cancer? There are many applications and many more remaining questions. We hope, however, that the work presented here will provide both the foundation and the motivation to explore these many avenues.

Author Contributions: Conceptualization, P.J.G. and I.V.-M.; methodology, P.J.G., L.M.W., E.E.R.-T., M.E. and I.V.-M.; software, I.V.-M. and M.E.; validation, I.V.-M., M.E.; formal analysis, P.J.G. and I.V.-M.; resources, P.J.G., L.M.W., E.E.R.-T.; writing—original draft preparation, P.J.G. and I.V.-M.; writing—review and editing, P.J.G., L.M.W., E.E.R.-T., M.E. and I.V.-M.; supervision, P.J.G. and L.M.W.; project administration, P.J.G.; funding acquisition, P.J.G. and L.M.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the U.S. National Institutes of Health (National Institute of General Medical Sciences) grant number R35GM137919 (PI: G. Bradburd). I.V.M. would like to acknowledge the support of Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) through the scholarship for graduate studies CVU number 1177174; additionally, I.V.M. received

financial support from the Society for the Study of Evolution to attend and present this work at the Evolution Meeting 2023 in Albuquerque, NM, enhancing the project's scope. L.M.W. gratefully acknowledges the support of the Natural Sciences and Engineering Research Council of Canada grant RGPIN-2019-06294.

Data Availability Statement: No data were used. An R package developed by I.V. for analyses of real data is provided at: <https://github.com/isaacvazquez1/EstimatingMutationRates>, accessed on 31 January 2024.

Acknowledgments: We would like to thank the Center for Advanced Research Computing at the University of New Mexico, supported in part by the National Science Foundation, for providing the high-performance computing resources used in this work. P.G. received financial support from the USA/Brazil Fulbright scholar program. P.G. received further support from NIH (NIGMS) grant number R35GM137919 (PI: G. Bradburd).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MLE maximum likelihood estimation
pgf probability-generating function
pmf probability mass function

Appendix A. Simulations

Appendix A.1. Graphical Method

Here, we describe a simple rejection method, based on a graphical representation of mutations on a plane, that greatly reduces computation time when the final population size is very large.

Suppose we grow a bacterial population inside a flask, where diet and space requirements are fulfilled for up to N cells and no cell dies. Taking N as the carrying capacity and considering a constant per-generation growth rate $r = \log 2$ associated with each individual in the bacterial population, we have a pure-birth process where the reproduction of each member acts independently of the others, i.e., we have a stochastic process called the Yule process, a continuous-time homogeneous Markov chain [39].

In this continuous-time setting, the population size reaches the carrying capacity N in t_f generations and the population grows no further. Here, as in many biological contexts, time is measured in generations; i.e., $t_f \in \mathbb{R}$ is such that $2^{t_f} = N$. Additionally, let us suppose μ_m is the mutation rate to mutator phenotypes and μ' is the mutation rate of mutator cells to the marker phenotype.

With the above assumptions, the expected number of mutations to the mutator phenotype occurring from the beginning of the experiment until time t_f is

$$N t_f \mu_m, \quad (\text{A1})$$

according to [23] (Equation (6)). Moreover, if M is a random variable modeling the number of mutations occurring in the time interval $[0, t_f]$, M follows a Poisson distribution by the law of small numbers theorem [40] used as in [23], so

$$M \sim \text{Poisson}(N t_f \mu_m). \quad (\text{A2})$$

The Yule process stochastically characterizes the lineage size of cells as individuals, e.g., see Equation (A4). This notwithstanding, our framework leads to the following simplification: bacteria as a whole population grow deterministic as an exponential function; this is, given $t \in [0, t_f]$, then e^{rt} , with $r = \frac{\log(N)}{t_f}$, denotes the population size at time t .

Let $f: [0, t_f] \rightarrow [0, 1]$ be the function such that

$$f(t) = \frac{e^{rt}}{N}. \quad (\text{A3})$$

Hence, $f(t)$ denotes the population size at time t compared to the final population, as a fraction.

Let $C = (T, F)$ be a random vector, where $T \sim \text{Uniform}(0, t_f)$ and $F \sim \text{Uniform}(0, 1)$, then C represents a cell birthed at time T with a mutation or, equivalently, a mutant cell that appeared at time T and belongs to the fraction F of the final population. By considering a random sample of size M from C , $C_1 = (T_1, F_1), C_2 = (T_2, F_2), \dots, C_M = (T_M, F_M)$ and scattering them within $[0, t_f] \times [0, 1]$, we obtain feasible and not feasible mutants, since, regarding f as a probability density function, all feasible mutations lie under the graph of f , and disregard them otherwise.

Figure A1 shows an example of the above-described method.

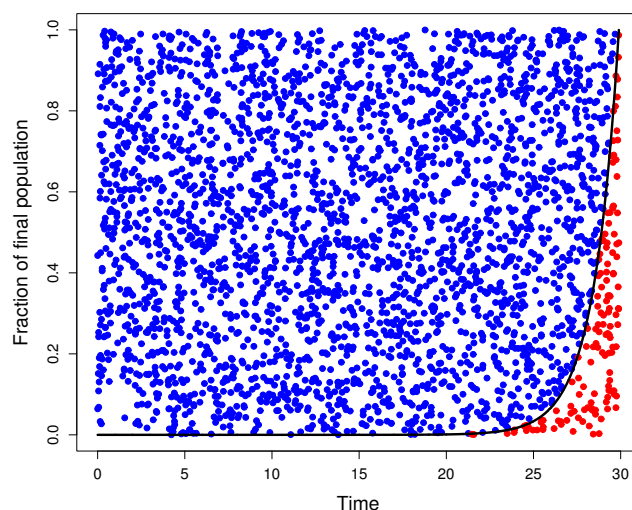


Figure A1. Graphical model for selecting feasible mutations. $M \sim \text{Poisson}(Nt_f\mu_m)$ mutations were generated as random points $C_1, C_2, \dots, C_M$ on $[0, t_f] \times [0, 1]$. The parameter for the final population size was $N = 2^{33}$, for the wild-type mutation rate was $\mu_m = 10^{-7}$, and for the final time was $t_f = \log_2(N) \approx 29.89735$. The graph of $f(t) = \frac{1}{N} \exp\left(\frac{\log(N)t}{t_f}\right)$ (black curve) played a discriminant role since all points below the graph of f (red points) represent actual mutations, while the rest of the points (blue points) were disregarded.

Let M_m be the number of actual mutations from wild-type to mutator cells. Without loss of generality, let us suppose $C_i = (T_i, F_i)$ is the mutant cell associated with the i^{th} mutation, $i = 1, \dots, M_m$. Let us note that each mutant cell C_i , appearing at time T_i , has $t_f - T_i$ units of time to give rise to its clonal lineage, i.e., a subpopulation of size $X_i \geq 1$. Now, in a Yule process, the clone population size that starts with $j > 0$ individuals, each one with growth rate λ , has a negative binomial (NB) distribution with parameters j and $e^{-\lambda t}$ ([39] (pp. 377–378); [41] (Equation 3.15); [42]).

Therefore, X_i is a random variable such that

$$X_i - 1 \sim \text{NB}\left(1, e^{-r(t_f - T_i)}\right), \forall i \in \{1, \dots, M_m\}, \quad (\text{A4})$$

since $X_i - 1$ denotes the number of cells in the lineage of cell C_i , excluding the initial cell C_i . This is due to the support of the negative binomial distribution we are computationally working with [43] (func. `rnbino`), $\{0, 1, 2, \dots\}$. When the support of this distribution is $\{1, 2, \dots\}$, as in ([39] (pp. 377–378); [41] (Equation 3.15); [42]), Equation (A4) should be rewritten as

$$X_i \sim \text{NB}\left(1, e^{-r(t_f - T_i)}\right), \forall i \in \{1, \dots, M_m\}.$$

Furthermore, in both cases, the total number of mutators in the population at time t_f is

$$X = \sum_{i=1}^{M_m} X_i. \quad (\text{A5})$$

To incorporate mutant cells with mutator phenotypes, let us suppose they appear as the result of mutations within the mutator cells' subpopulation. Then, mutant cells with mutator phenotypes can appear only along the growth cycle of each mutant cell C_i , $i = 1, \dots, M_m$.

Let $i \in \{1, \dots, M_m\}$ be a fixed index. We know there are exactly X_i mutator cells that have been raised by mutator cell $C_i = (T_i, F_i)$, and if any mutant cell with a mutator phenotype, i.e., a mutator cell with a marker mutation, is appearing will be in the time interval $[T_i, t_f]$. Setting X_i as the carrying capacity, mutant cells with mutator phenotypes have $t_f - T_i$ units of time to appear, which happens with frequency μ' .

Consequently, under a similar reasoning for Equation (A2), we have that the number of mutations leading to mutant cells with mutator phenotypes is a random variable M_i such that

$$M_i \sim \text{Poisson}(X_i(t_f - T_i)\mu'). \quad (\text{A6})$$

Let $C_{i,1} = (T_{i,1}, F_{i,1}), \dots, C_{i,M_i} = (T_{i,M_i}, F_{i,M_i})$, with $T_{i,j} \sim \text{Uniform}(0, t_f - T_i)$ and $F_{i,j} \sim \text{Uniform}(0, 1)$, a random sample of size M_i . Hence, each $C_{i,j}$, $j \in \{1, \dots, M_i\}$ is a random point on $[0, T_i] \times [0, 1]$, representing a potential mutant cell with a mutator phenotype, i.e., a mutator cell with a marker mutation.

Analogous to Equation (A3), function $f_i: [0, t_f - T_i] \rightarrow [0, 1]$ defined by

$$f_i(t) = \frac{e^{r_i t}}{X_i}, \quad (\text{A7})$$

with $r_i = \frac{\log(X_i)}{t_f - T_i}$, plays the discriminant role for the M_i mutant cells with mutator phenotypes, where, as before, all points except those below the graph of f_i were disregarded since they represent actual mutant cells with mutator phenotypes.

Let $M_{m,i}$ be the number of actual mutant cells with mutator phenotypes that appeared along the growth cycle of mutator cell C_i . Without loss of generality, let us say $C_{i,1}, \dots, C_{i,M_{m,i}}$ represent mutant cells with mutator phenotypes, then, for each $j \in \{1, \dots, M_{m,i}\}$, the mutant cell with a mutator phenotype $C_{i,j}$ has $t_f - (T_i + T_{i,j})$ units of time to give rise to its offspring of size $X_{i,j}$, which, according to Equation (A4), $X_{i,j}$ is a random variable such that

$$X_{i,j} - 1 \sim \text{NB}\left(1, e^{-r_i(t_f - T_i - T_{i,j})}\right), \quad (\text{A8})$$

for every $j \in \{1, \dots, M_{m,i}\}$.

By repeating this process for each $i \in \{1, \dots, M_m\}$, we have that the total number of mutators in the population at time t_f is

$$Y = \sum_{i=1}^{M_m} \sum_{j=1}^{M_{m,i}} X_{i,j}. \quad (\text{A9})$$

Furthermore, we can study the distribution of the numbers of mutator cells and mutant cells with mutator phenotypes, as shown in Figure A2.

For a pseudocode description of the graphical model, see Algorithm A1.

Algorithm A1: Graphical model pseudocode**Input:**

- N : Final population size.
- μ_m : Mutation rate to mutator phenotypes.
- μ' : Mutation rate to mutants with mutator phenotypes.
- t_f : Final time.

```

1  $g_r = \log(N)t_f^{-1}$ 
2  $M$  a random sample of size one from  $M \sim \text{Poisson}(Nt_f\mu_m)$ .
3  $T$  array of a random sample of size  $M$  from  $T \sim \text{Uniform}(0, t_f)$ .
4  $P$  array of a random sample of size  $M$  from  $P \sim \text{Uniform}(0, 1)$ .
5  $C$  an array of couples  $(T[i], P[i])$  such that  $P[i] < N^{-1} \exp(T[i]g_r)$ .
6 actual_mutations number of couples in  $C$ .
7 time_difference  $1 \times \text{actual\_mutations}$  array.
8 Mutators  $1 \times \text{actual\_mutations}$  array.
9 Total_mutators = 0.
10 Total_mutant_mutators = 0.
11 for  $i=1$  to actual_mutations by steps of 1 do
12   time_difference[i] =  $t_f - C[1, i]$ .
13   Mutators[i] = 1+nb, nb a a random sample of size one from
      $X \sim \text{NB}(1, \exp(-g_r \text{time\_difference}[i]))$ .
14   Total_mutators = Total_mutators + Mutators[i].
15    $g_{ri} = \log(\text{Mutators}[i]) (\text{time\_difference}[i])^{-1}$ .
16    $M_i$  a random sample of size one from
      $M_i \sim \text{Poisson}(\text{Mutators}[i] \text{time\_difference}[i] \mu')$ .
17    $T_i$  array of a random sample of size  $M_i$  from
      $T_i \sim \text{Uniform}(0, t_f - \text{time\_difference}[i])$ .
18    $P_i$  array of a random sample of size  $M_i$  from  $P_i \sim \text{Uniform}(0, 1)$ .
19    $C_i$  an array of couples  $(T_{ij}, P_{ij})$  such that
      $P_{ij} < (\text{Mutators}[i])^{-1} \exp(T_{ij}g_{ri})$ .
20   actual_mutations_i number of couples in  $C_i$ .
21   time_difference_i  $1 \times \text{actual\_mutations\_i}$  array.
22   Mutant_mutators  $1 \times \text{actual\_mutations\_i}$  array.
23   Total_mutant_mutators_i = 0.
24   for  $j=1$  to actual_mutations_i by steps of 1 do
25     time_difference_i[j] = time_difference[i] -  $C_i[1, j]$ .
26     Mutant_mutators[j] = 1+nb_i, nb_i a random sample of size one from
        $X_i \sim \text{NB}(1, \exp(-g_{ri} \text{time\_difference\_i}[j]))$ .
27     Total_mutant_mutators_i = Total_mutant_mutators_i + Mutant_mutators[j].
28   Total_mutant_mutators = Total_mutant_mutators + Total_mutant_mutators_i.

```

Output: Total_mutators; Total_mutant_mutators.

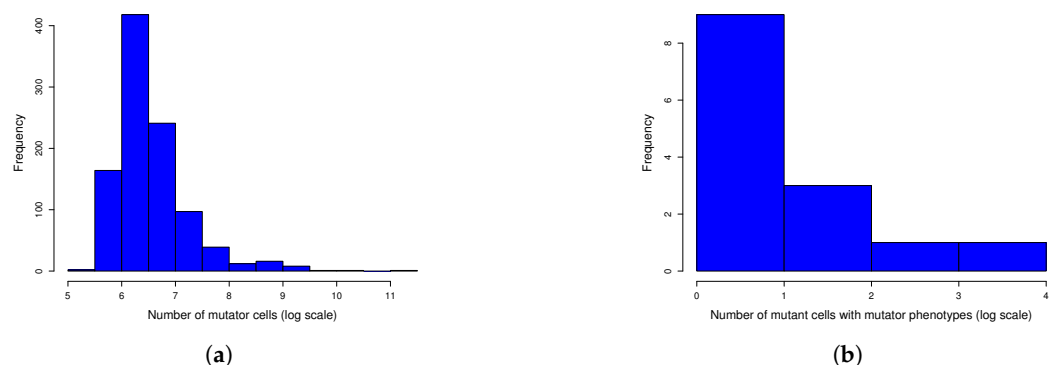


Figure A2. One thousand simulations of graphical method. (a) Histogram of the number of mutants on log scale and (b) histogram of the number of mutators on log scale. Given the parameters in Figure A1 and a mutation rate to mutator phenotype $\mu' = 10^{-5}$, we simulated 1000 cultures in 354 s (printing plots) and in 5 s (printing no plots) of the described method. The distributions of the numbers of mutants and mutators were heavy-tailed, as expected.

Appendix A.2. Quantile Function Model

Let us recall that, given a function $g: A \rightarrow B$, we say it is invertible if there exists a function $g^{-1}: B \rightarrow A$ (called the inverse function of g) such that

$$\begin{aligned} g^{-1}(g(a)) &= a, \forall a \in A, \text{ and} \\ g(g^{-1}(b)) &= b, \forall b \in B. \end{aligned}$$

Definition A1 (Quantile function). Let $F: \mathbb{R} \rightarrow [0, 1]$ be a cumulative distribution function. The quantile function $F^{-1}: [0, 1] \rightarrow [-\infty, \infty]$ of F is defined by

$$F^{-1}(x) = \inf\{y \in \mathbb{R}: F(y) \geq x\}, \quad (\text{A10})$$

$$0 \leq x \leq 1.$$

Quantile functions generalize the concept of the inverse of a function g when g is a cumulative distribution function. Now, quantile functions have a remarkable application, often called the inversion method, which is often employed to simulate random variables [44] (Proposition 2), as will be used in the following method.

Under the assumptions and using the notation of the graphical model, given $t \in [0, t_f]$, e^{rt} denotes the population size at time t . Then, $\mu_m e^{rt}$ denotes the expected number of mutant cells at time t , which, as a function, can be regarded as a probability density function. Furthermore, let $k = r/(N - 1)\mu_m$, then $h: [0, t_f] \rightarrow [k\mu_m, k\mu_m N]$, given by

$$h(t) = k\mu_m e^{rt} \quad (\text{A11})$$

is a probability density function, where its cumulative distribution function is $H: [0, t_f] \rightarrow [0, 1]$ defined by

$$H(t) = k\mu_m \frac{e^{rt} - 1}{r}. \quad (\text{A12})$$

Hence, its quantile function is $H^{-1}: [0, 1] \rightarrow [0, t_f]$ given by

$$H^{-1}(u) = \frac{1}{r} \log\left(\frac{ur}{k\mu_m} + 1\right). \quad (\text{A13})$$

So, if $U \sim \text{Uniform}(0, 1)$, then

$$\begin{aligned} V &= \frac{1}{r} \log \left(\frac{Ur}{k\mu_m} + 1 \right) \\ &= \frac{1}{r} \log(U(N-1) + 1) \end{aligned} \quad (\text{A14})$$

is such that $V \sim H$, by the inversion method.

Applying the inversion method as above, only actual mutations leading to mutator phenotypes and their respective time of appearance are provided (see Figure A3). Thus, instead of asking for the expected number of mutations, as in Equation (A1), it is sufficient to ask for the expected number of mutators in the population, which is

$$N\mu_m. \quad (\text{A15})$$

Therefore, Equation (A2) is rewritten as

$$M \sim \text{Poisson}(N\mu_m) \quad (\text{A16})$$

Then, we can get the time of appearance of M mutator cells by generating a random sample from V of size M , $V_1, \dots, V_M$. Note that, with this framework, there is no need to disregard any point. Therefore, the number of actual mutations M_m coincides with M , which implies Equation (A4) is rewritten as

$$X_i - 1 \sim \text{NB} \left(1, e^{-r(t_f - V_i)} \right), \forall i \in \{1, \dots, M\}, \quad (\text{A17})$$

and then, the total number of mutators in the population at time t_f is

$$X = \sum_{i=1}^M X_i. \quad (\text{A18})$$

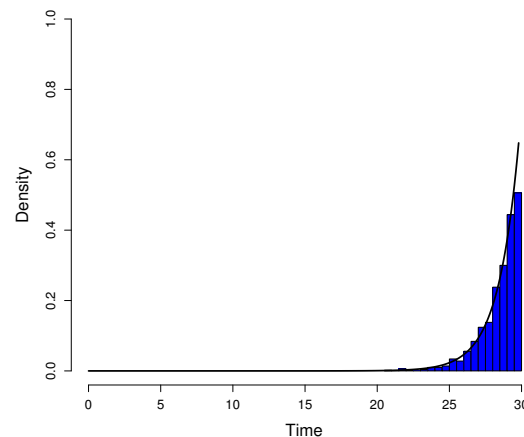


Figure A3. Comparison between simulated data and actual probability density function. The parameters were the same as in Figure A1. Data for the histogram of the time of appearance of mutant cells were simulated by the inversion method and Equation (A14). The graph of $h(t) = k\mu_m e^{rt}$ (black curve) was included for reference since the simulated data have distribution h .

To incorporate mutant cells with mutator phenotypes, we proceed as in Equation (A15). For each $i \in \{1, \dots, M\}$, the expected number of mutants within the i^{th} mutator's offspring of size X_i is $X_i\mu'$. As a consequence, the random variable, M_i , counting the number of mutations leading to mutant cells with the i^{th} mutator phenotype, is such that

$$M_i \sim \text{Poisson}(X_i\mu'). \quad (\text{A19})$$

Now, given $i \in \{1, \dots, M\}$, a fixed index, let $k_i = r_i / (X_i - 1)\mu'$, then $h_i: [0, t_f - V_i] \rightarrow [k_i\mu', k_i\mu'X_i]$, defined by

$$h_i(t) = k_i\mu' e^{r_i t} \quad (\text{A20})$$

is the probability density function whose quantile function $H_i^-: [0, 1] \rightarrow [0, t_f - V_i]$ is given by

$$H_i^-(u) = \frac{1}{r_i} \log\left(\frac{ur_i}{k_i\mu'} + 1\right). \quad (\text{A21})$$

Thus, if $U_i \sim \text{Uniform}(0, 1)$, then

$$\begin{aligned} W_i &= \frac{1}{r_i} \log\left(\frac{U_i r_i}{k_i \mu'} + 1\right) \\ &= \frac{1}{r_i} \log(U_i (X_i - 1) + 1) \end{aligned} \quad (\text{A22})$$

is such that $W_i \sim H_i$, where $H_i(t) = \int_0^t h_i(\tau) d\tau$.

Let $W_{i,1}, \dots, W_{i,M_i}$ be a random sample from W_i , then $V_i + W_{i,j}$ denotes the time of appearance of the j^{th} mutant cell with the i^{th} mutator phenotype. Therefore, analogous to Equation (A17), the size of their offspring is a random variable $X_{i,j}$ such that

$$X_{i,j} - 1 \sim \text{NB}\left(1, e^{-r_i(t_f - V_i - W_{i,j})}\right), \forall j \in \{1, \dots, M_i\}. \quad (\text{A23})$$

This implies that the number of mutant cells with mutator phenotypes is

$$Y = \sum_{i=1}^M \sum_{j=1}^{M_i} X_{i,j}. \quad (\text{A24})$$

For a pseudocode description of the quantile function method, refer to Algorithm A2.

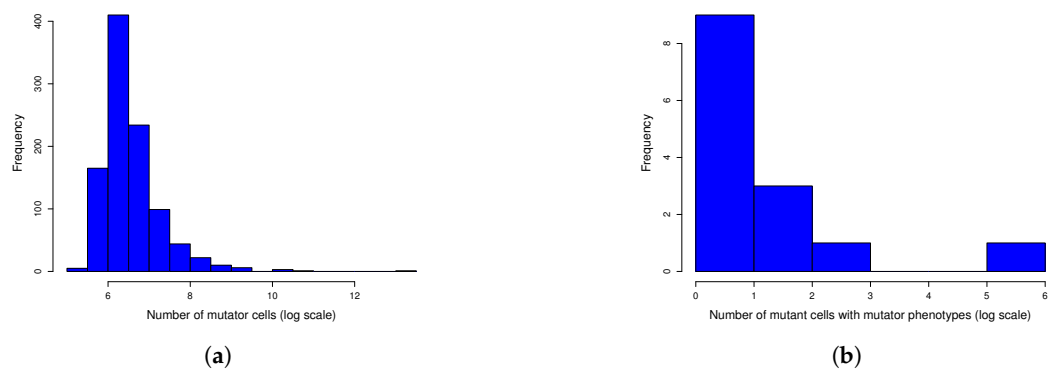


Figure A4. (a) Histogram of the number of mutants on log scale and (b) histogram of the number of mutators on log scale. The same parameters of Figure A2 were used to simulate 1000 cultures in two seconds using the quantile function method.

Appendix A.3. Description of the Pure-Birth Process Simulation

We started the simulation with a single individual referred to as a normal cell (neither a mutator, nor antibiotic-resistant). In order to simulate a pure-birth process, a single individual was chosen at random from the population to reproduce. This individual divides into two equal daughter cells, and the mother cell is removed from the population.

At each birth event, each normal daughter cell may mutate to the antibiotic-resistant phenotype at mutation rate μ or to the mutator phenotype at rate μ_m . Individuals with the

mutator phenotype have a mutation rate that is k -times higher than the normal mutation rate. When either mutation happens, all subsequent offspring inherit the new phenotype.

We note that the mutator phenotype does not have antibiotic resistance directly, but its offspring can develop antibiotic resistance in the future. In particular, daughters of a mutator cell develop antibiotic resistance with mutation rate $\mu' = k\mu$, while daughters of an antibiotic resistant cell become mutators at rate μ_m . Consequently, the population consists of four types of individuals: normal cells, antibiotic resistant cells, mutator cells, and individuals exhibiting both antibiotic resistance and the mutator phenotype.

Algorithm A2: Quantile function model pseudocode

Input:

- N : Final population size.
- μ_m : Mutation rate to mutator phenotypes.
- μ' : Mutation rate to mutants with mutator phenotypes.
- t_f : Final time.

```

1  $g_r = \log(N)t_f^{-1}$ 
2  $M$  a random sample of size one from  $M \sim \text{Poisson}(N\mu_m)$ .
3  $U$  a random sample of size  $M$  from  $U \sim \text{Uniform}(0,1)$ .
4  $V$   $1 \times M$  array.
5  $\text{time\_difference}$   $1 \times M$  array.
6  $\text{Mutators}$   $1 \times M$  array.
7  $\text{Total\_mutators} = 0$ .
8  $\text{Total\_mutant\_mutators} = 0$ .
9 for  $i = 1$  to  $M$  by steps of 1 do
10    $V[i] = g_r^{-1} \log(U[i](N-1) + 1)$ .
11    $\text{time\_difference}[i] = t_f - V[i]$ .
12    $\text{Mutators}[i] = 1 + \text{nb}$ ,  $\text{nb}$  a random sample of size one from
      $X \sim \text{NB}(1, \exp(-g_r \text{time\_difference}[i]))$ .
13    $\text{Total\_mutators} = \text{Total\_mutators} + \text{Mutators}[i]$ .
14    $g_{ri} = \log(\text{Mutators}[i]) (\text{time\_difference}[i])^{-1}$ .
15    $M_i$  a random sample of size one from  $M_i \sim \text{Poisson}(\text{Mutators}[i] \mu')$ .
16    $U_i$  a random sample of size  $M_i$  from  $U_i \sim \text{Uniform}(0,1)$ .
17    $W_i$   $1 \times M_i$  array.
18    $\text{time\_difference}_i$   $1 \times M_i$  array.
19    $\text{Mutant\_mutators}$   $1 \times M_i$  array.
20    $\text{Total\_mutant\_mutators}_i = 0$ .
21   for  $j = 1$  to  $M_i$  by steps of 1 do
22      $\text{time\_difference}_i[j] = \text{time\_difference}[i] - W_i[j]$ .
23      $\text{Mutant\_mutators}[j] = 1 + \text{nb}_i$ ,  $\text{nb}_i$  a random sample of size one from
        $X_i \sim \text{NB}(1, \exp(-g_{ri} \text{time\_difference}_i[j]))$ .
24      $\text{Total\_mutant\_mutators}_i = \text{Total\_mutant\_mutators}_i + \text{Mutant\_mutators}[j]$ .
25    $\text{Total\_mutant\_mutators} = \text{Total\_mutant\_mutators} + \text{Total\_mutant\_mutators}_i$ .

```

Output: Total_mutators ; $\text{Total_mutant_mutators}$.

In summary, the simulation proceeds by choosing an individual at random to reproduce, removing that individual, and determining which types of daughters are added in its place, as described above. This process is repeated until the population size reaches the desired final size (e.g., 2^{33}). We can also simulate repeated growth phases and population bottlenecks by randomly sampling the final population to achieve a desired initial population size (e.g., 2^{20}) and repeating the procedure above. See Algorithms A3–A5 for a detailed pseudocode description.

Algorithm A3: Pure-birth simulation pseudocode

```

1 for ireps = 0 to nreps
    // Initialize Variables.
2   NAMB[0] = Ninit; NAMB[1]=0; NAMB[2]=0; NAMB[3]=0;
    // For loop for the growth phase.
3   for ireps = 0 to nreps
4       if igrowth > 0 then
5           // bottleneck before growing except for the first growth
           // phase.
           Set bottle =  $2^{p_{init}[igrowth]} / N_{total}$ ;
           // For loop for random sampling of individuals.
6           for i = 0 to 3 do
7               Set NAMB[i] = Binomial(bottle, NAMB[i]);
           // Compute the total population size.
8           Ntotal = NAMB[0]+NAMB[1]+NAMB[2]+NAMB[3];
           // Calculate cumulative frequency of each type.
           cumsum[0] = NAMB[0]/Ntotal;
9           for i = 1 to 2 do
10              cumsum[i] = cumsum[i-1] + NAMB[i]/Ntotal;
           // While loop for growth phase, only daughters mutate.
12          while Ntotal < N do
           // Choose which type reproduces based on random number.
13          if (r < cumsum[0]) then
           // Normal cell reproduces.
           // The mother cell is removed from the population.
14          NAMB[0] = NAMB[0]-1;
           // If this random number is greater than the sum of the
           // mutation rates for antibiotic resistance and mutator,
           // a normal cell is produced.
15          if ( $r_2 = r / \text{cumsum}[0] > \mu + \mu_m$ ) then
16              NAMB[0] = NAMB[0]+1;
           // If it is greater than the mutation rate for mutator,
           // the mutator reproduces.
17          else if ( $r_2 > \mu$ ) then
18              NAMB[2] = NAMB[2]+1;
           // Otherwise, the resistant type reproduces.
19          else
20              NAMB[1] = NAMB[1]+1;
           // This process is repeated for the second daughter cell
           // in the same pattern.
21          if ( $r_3 > \mu + \mu_m$ ) then
22              NAMB[0] = NAMB[0]+1;
23          else if ( $r_3 > \mu$ ) then
24              NAMB[2] = NAMB[2]+1;
25          else
26              NAMB[1] = NAMB[1]+1;

```

Algorithm A4: Pure-birth simulation definitions

Input:

- $\mu = 10^{-7}$: Mutation rate from wild-type to antibiotic resistance.
 - $\mu_m = 10^{-6}$: Mutation rate from wild-type to mutator.
 - $\mu' = 10^{-5}$: Mutation rate from mutator to antibiotic resistance.
 - $N = 2^{33}$: Final population size.
 - Ntotal: Current population size, total of all types.
 - NAMB[0], NAMB[1], NAMB[2], NAMB[3]: Numbers of each of four types of individuals including normal cells, antibiotic-resistant cells, mutator cells, and individuals including both antibiotic-resistant and mutator phenotype respectively.
 - nreps = 1000: Number of replicate lines.
 - cumsum: Cumulative frequency of each type in the population, set cumsum[3] = 1.
 - pinit = [0,20,20,20]: (see below)
 - Ninit: initial population size for each growth phase, $= 2^{pinit}$.
 - bottle: Bottleneck sampling fraction.
 - r, r_2, r_3 : Random numbers.
 - ngrowths = 4: number of growth phases.
-

Appendix B. Estimation Methods

In previous works dedicated to estimating the mutation rate from wild-type to mutant cells [45–52], methodologies with high accuracy have been developed. In this study, we relied on this literature, which is primarily based on the probability-generating function and its empirical counterpart [53]. Additionally, we discuss the methods we developed to estimate the mutation rate from wild-type to mutator phenotypes. The first method implements direct fitting techniques using the least-squares method, minimizing the *distance* between the probability-generating function and the empirical probability-generating function to produce the desired estimates. Meanwhile, the second method takes greater advantage of the dilution process proposed as part of the experimental protocol, utilizing the fast Fourier transform and maximum likelihood methods to generate the estimations.

Appendix B.1. Direct Estimations

Let X be a discrete real-valued random variable with support S , probability mass function $p(x) = \mathbb{P}(X = x)$, for all $x \in S$, and let G_X be the probability-generating function of X , defined as:

$$G_X(\theta) = \mathbb{E}(\theta^X) = \sum_{x \in S} p(x)\theta^x. \quad (\text{A25})$$

We note that G_X is defined on some region $R \subset \mathbb{C}$ and that R contains the unit disk $R_0 = \{\theta \in \mathbb{C}: |\theta| \leq 1\}$ because, for any arbitrary and fixed complex number θ such that $|\theta| \leq 1$, it holds that

$$\sum_{x \in S} |p(x)\theta^x| \leq \sum_{x \in S} p(x) = 1.$$

The comparison test implies G_X converges absolutely on R_0 , which proves the statement.

Algorithm A5: Pure-birth simulation pseudocode, continued

```

27 // Antibiotic-Resistant cell reproduces.
    else if  $r < cumsum[1]$  then
28     // The mother cell is removed from the population.
        NAMB[1] = NAMB[1]-1;
29     // Both daughter cells become antibiotic-resistant.
        if ( $r_2 > 2 \times \mu_m \times (1 - \mu_m) + \mu_m \times \mu_m$ ) then
30         | NAMB[1] = NAMB[1] + 2;
        // Only one of them becomes antibiotic-resistant while
        // the other becomes a mutator.
31     else if ( $r_2 > \mu_m \times \mu_m$ ) then
32         | NAMB[1] = NAMB[1]+1; NAMB[3] = NAMB[3]+1;
        // Otherwise, both cells become an individual exhibiting
        // antibiotic-resistant and mutator.
33     else
34         | NAMB[3] = NAMB[3] + 2;
    // Mutator cell reproduces.
35    else if  $r < cumsum[2]$  then
        // The mother cell is removed from the population.
36        NAMB[2] = NAMB[2]-1;
        // Both daughter cells become mutator.
37        if ( $r_2 > 2 \times \mu' \times (1 - \mu') + \mu' \times \mu'$ ) then
38            | NAMB[2] = NAMB[2] + 2;
        // Only one of them becomes mutator while the other
        // becomes resistant mutator.
39        else if ( $r_2 > \mu' \times \mu'$ ) then
40            | NAMB[2]=NAMB[2]+1; NAMB[3]=NAMB[3]+1;
        // Otherwise, both cells become resistant mutators.
41        else
42            | NAMB[3] = NAMB[3] + 2;
    // Otherwise both daughter cells become resistant mutators.
43    else
44        | NAMB[3] = NAMB[3]+1;
45    Ntotal = Ntotal + 1;

```

However, when working with the Luria–Delbrück probability-generating function, it is customary to consider the set $\{\theta \in \mathbb{R}: 0 \leq \theta \leq 1\}$ exclusively; i.e., the interval $[0, 1]$. This is primarily due to biological considerations. Values close to zero, approaching from the right side, reflect the effect of observing small quantities of mutant cells, while values close to one, approaching from the left side, reflect the influence of large quantities of mutant cells in cell cultures [24].

In works such as [24,31], the study of the probability-generating function has been restricted to the points $\theta \in \Theta$, where $\Theta = \{0.1, 0.8, 0.9\}$, to generate estimates of μ . However, to estimate μ_m , we considered the points $\theta \in \Theta^*$, where $\Theta^* = \{0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$.

This choice was made because we will be using the probability-generating function for the sum of the random variable counting the number of mutant cells produced by wild-type cells and the one counting the number of mutant cells produced by mutator cells. Since we expect to observe small quantities of mutant cells sporadically, considering the set Θ^* allowed us to reflect this framework.

Let us assume that r^* replicates of the experiment have been considered, each with c^* growth cycles. Let $m_i \in \mathbb{R}^{r^*}$, where $i = 1, \dots, c^*$, be a vector whose j^{th} entry, $j = 1, \dots, r^*$, represents the sum of the number of mutant cells produced by wild-type cells and the number of mutant cells produced by mutator cells.

To incorporate this information into the dilution process of the experimental protocol, let us define the vector $n_i \in \mathbb{R}^{r^*}$, where $i = 1, \dots, c^*$, such that its j^{th} entry, $j = 1, \dots, r^*$, corresponds to a realization of the random variable S_j

$$S_j \sim \text{Binomial}(m_{i,j}, \delta_i),$$

where $m_{i,j}$ and $\delta_i \in (0, 1)$ represent the j^{th} entry of the vector m_i and the dilution factor considered for the i^{th} growth cycle, respectively.

Before introducing the estimation method, we propose intervals $I = [l, u]$, $I' = [l', u']$, and $I_m = [l_m, u_m]$ such that

$$\begin{aligned} \log_{10}(\mu) &\in I, \\ \log_{10}(\mu') &\in I', \text{ and} \\ \log_{10}(\mu_m) &\in I_m. \end{aligned}$$

For computational purposes, the approach above provides greater accuracy in the estimation process.

To work with a finite number of possibilities, the intervals will be partitioned to consider only $(p^* + 1)^3$ points within the set $\Omega := I \times I' \times I_m$. Thus, instead of working with the interval $[a, b]$, where $(a, b) \in \{(l, u), (l', u'), (l_m, u_m)\}$, we will work with the set of points

$$\left\{ a \left(\frac{b}{a} \right)^{q/p^*} : q = 0, 1, \dots, p^* \right\} \subset [a, b]. \quad (\text{A26})$$

Let $F_1: \Theta^* \times \Omega \rightarrow \mathbb{R}$ be defined as

$$F_1(\theta, \omega_1, \omega_2, \omega_3) = h(\theta)^{\omega_1} h(h(\theta)^{\omega_2})^{\omega_3}, \quad (\text{A27})$$

and let $F_i: \Theta^* \times \Omega \rightarrow \mathbb{R}$ be given by

$$F_i(\theta, \omega_1, \omega_2, \omega_3) = h(F_{i-1}(\theta))^{\omega_1} h(h(F_{i-1}(\theta))^{\omega_2})^{\omega_3}, \quad (\text{A28})$$

for $i = 2, \dots, c^*$.

In this way, the estimators $\hat{\Delta}_i$, $\hat{\mu}'_i$, and $\hat{\Delta}_{mi}$ for Δ , μ' , and Δ_m in the i^{th} growth cycle, respectively, are those that satisfy

$$\sum_{\theta \in \Theta^*} \left(F_i(\theta, \hat{\Delta}_i, \hat{\mu}'_i, \hat{\Delta}_{mi}) - G_i(\theta) \right)^2 = \min_{(\omega_1, \omega_2, \omega_3) \in \Omega} \sum_{\theta \in \Theta^*} \left(F_i(\theta, N10^{\omega_1}, 10^{\omega_2}, N10^{\omega_3}) - G_i(\theta) \right)^2,$$

for $i = 1, \dots, c^*$.

Appendix B.2. Maximum Likelihood Estimations

Applying maximum likelihood methods requires working directly with probability mass functions. In this case, it would involve dealing with the explicit expression of the Luria–Delbrück distribution, which admits no explicit analytic expression [33].

However, by considering the direct approach, outlined up to Equation (A28), sufficient information is provided to recover the probability mass states of the Luria–Delbrück distribution and implement maximum likelihood methods. This can be done by employing the fast Fourier transform as in [54,55] (Equation (5)).

The Luria–Delbrück distribution, in its simplest form (i.e., the Lea–Coulson formulation that we employ here), has no finite moments; hence, its moment-generating function is not analytic. Its characteristic function, on the other hand, is analytic, and it is $F(e^{i\theta})$. We leveraged this fact to obtain the probability mass function defined by the inverse Fourier transform of $F(e^{i\theta})$.

Let $d_i = \max_{j \in \{1, \dots, r^*\}} n_{ij}$, where n_{ij} is the j^{th} entry of the vector n_i , for $i = 1, \dots, r^*$.

In other words, d_i is the maximum number of mutant cells present in the i^{th} growth cycle after dilution.

In this way, given $\omega = (\omega_1, \omega_2, \omega_3) \in \Omega$ and defining $\mathcal{F}_i(\theta; \omega) := F_i(\theta, N10^{\omega_1}, 10^{\omega_2}, N10^{\omega_3})$, the real part of the fast Fourier transform of $\mathcal{F}_i$:

$$\check{\mathcal{F}}_i(k) = \frac{1}{d_i} \sum_{l=0}^{d_i-1} \mathcal{F}_i\left(\frac{2\pi i l}{d_i}; \omega\right) \exp\left(-\frac{2\pi i l k}{d_i}\right), \quad (\text{A29})$$

where $i^2 = -1$, approximates the value of

$$p_{X_i}(k; \omega) = P(X_i = k; \omega), \quad (\text{A30})$$

for $k = 0, 1, \dots, d_i$, where X_i is the random variable whose probability-generating function is $\mathcal{F}_i$. In other words, Equation (A29) provides an approximation to the probability of observing k mutants on the i^{th} growth cycle after the dilution.

Therefore,

$$\mathcal{L}_i(\omega) = \prod_{j=1}^{r^*} p_{X_i}(n_{ij}; \omega)$$

denotes the likelihood function associated with the i^{th} growth cycle.

This implies that the maximum likelihood estimators $\hat{\Delta}_i^*$, $\hat{\mu}'_i$, and $\hat{\Delta}_{m_i}^*$ for Δ , μ' , and Δ_m in the i^{th} growth cycle, respectively, are those satisfying

$$\mathcal{L}_i(\hat{\Delta}_i^*, \hat{\mu}'_i, \hat{\Delta}_{m_i}^*) = \max_{\omega \in \Omega} \mathcal{L}_i(\omega). \quad (\text{A31})$$

References

1. Drake, J.W. A constant rate of spontaneous mutation in DNA-based microbes. *Proc. Natl. Acad. Sci. USA* **1991**, *88*, 7160–7164. [CrossRef] [PubMed]
2. Drake, J.W.; Charlesworth, B.; Charlesworth, D.; Crow, J.F. Rates of spontaneous mutation. *Genetics* **1998**, *148*, 1667–1686. [CrossRef] [PubMed]
3. Lynch, M. Evolution of the mutation rate. *Trends Genet.* **2010**, *26*, 345–352. [CrossRef] [PubMed]
4. Sniegowski, P.D.; Gerrish, P.J.; Johnson, T.; Shaver, A. The evolution of mutation rates: Separating causes from consequences. *Bioessays* **2000**, *22*, 1057–1066. [CrossRef] [PubMed]
5. Sniegowski, P.D.; Gerrish, P.J.; Lenski, R.E. Evolution of high mutation rates in experimental populations of *E. coli*. *Nature* **1997**, *387*, 703–705. [CrossRef] [PubMed]
6. Yamagishi, J.; Yoshida, H.; Yamayoshi, M.; Nakamura, S. Nalidixic acid-resistant mutations of the *gyrB* gene of *Escherichia coli*. *Mol. Gen. Genet.* **1986**, *204*, 367–373. [CrossRef]
7. Mao, E.F.; Lane, L.; Lee, J.; Miller, J.H. Proliferation of mutators in a cell population. *J. Bacteriol.* **1997**, *179*, 417–422. [CrossRef]
8. Neinavaie, F.; Kramer, A. Does mutation rate of cancer cells change as the stage of the disease advances? *Cancer Res.* **2022**, *82*, A038. [CrossRef]
9. Hao, D.; Wang, L.; Di, L.J. Distinct mutation accumulation rates among tissues determine the variation in cancer risk. *Sci. Rep.* **2016**, *6*, 1–5. [CrossRef]
10. Tomlinson, I.P.M.; Novelli, M.R.; Bodmer, W.F. The mutation rate and cancer. *Proc. Natl. Acad. Sci. USA* **1996**, *93*, 14800–14803. [CrossRef]
11. Fox, E.J.; Prindle, M.J.; Loeb, L.A. Do mutator mutations fuel tumorigenesis? *Cancer Metastasis Rev.* **2013**, *32*, 353–361. [CrossRef] [PubMed]

12. Russo, M.; Pompei, S.; Sogari, A.; Corigliano, M.; Crisafulli, G.; Puliafito, A.; Lamba, S.; Erriquez, J.; Bertotti, A.; Gherardi, M.; et al. A modified fluctuation-test framework characterizes the population dynamics and mutation rate of colorectal cancer persister cells. *Nat. Genet.* **2022**, *54*, 976–984. [[CrossRef](#)] [[PubMed](#)]
13. Bielas, J.H.; Loeb, K.R.; Rubin, B.P.; True, L.D.; Loeb, L.A. Human cancers express a mutator phenotype. *Proc. Natl. Acad. Sci. USA* **2006**, *103*, 18238–18242. [[CrossRef](#)]
14. Natali, F.; Rancati, G. The Mutator Phenotype: Adapting Microbial Evolution to Cancer Biology. *Front. Genet.* **2019**, *10*, 713. [[CrossRef](#)] [[PubMed](#)]
15. Nowell, P.C. The clonal evolution of tumor cell populations. *Science* **1976**, *194*, 23–28. [[CrossRef](#)]
16. Sprouffske, K.; Merlo, L.M.F.; Gerrish, P.J.; Maley, C.C.; Sniegowski, P.D. Cancer in Light of Experimental Evolution. *Curr. Biol.* **2012**, *22*, R762–R771. [[CrossRef](#)]
17. Zheng, Q. A note on plating efficiency in fluctuation experiments. *Math. Biosci.* **2008**, *216*, 150–153. [[CrossRef](#)]
18. Gerrish, P. A simple formula for obtaining markedly improved mutation rate estimates. *Genetics* **2008**, *180*, 1773–1778. [[CrossRef](#)]
19. Jones, M.E.; Thomas, S.M.; Rogers, A. Luria-Delbrück fluctuation experiments: Design and analysis. *Genetics* **1994**, *136*, 1209–1216. [[CrossRef](#)]
20. Jones, M.E. An algorithm accounting for plating efficiency in estimating spontaneous mutation rates. *Comput. Biol. Med.* **1993**, *23*, 455–461. [[CrossRef](#)]
21. Stewart, F.M. Fluctuation analysis: The effect of plating efficiency. *Genetica* **1991**, *84*, 51–55. [[CrossRef](#)]
22. Bokes, P.; Singh, A. A modified fluctuation test for elucidating drug resistance in microbial and cancer cells. *Eur. J. Control* **2021**, *62*, 130–135. [[CrossRef](#)]
23. Luria, S.E.; Delbrück, M. Mutations of Bacteria from Virus Sensitivity to Virus Resistance. *Genetics* **1943**, *28*, 491–511. [[CrossRef](#)]
24. Hamon, A.; Ycart, B. Statistics for the Luria-Delbrück distribution. *EJSS* **2012**, *6*, 1251–1272. [[CrossRef](#)]
25. Ycart, B. Fluctuation analysis: Can estimates be trusted? *PLoS ONE* **2013**, *8*, e80958. [[CrossRef](#)] [[PubMed](#)]
26. Feller, W. *An Introduction to Probability Theory and Its Applications*; New York; Chapman & Hall: London, UK, 1957.
27. Lea, D.E.; Coulson, C.A. The distribution of the numbers of mutants in bacterial populations. *J. Genet.* **1949**, *49*, 264–285. [[CrossRef](#)] [[PubMed](#)]
28. Zheng, Q. Progress of a half century in the study of the Luria-Delbrück distribution. *Math. Biosci.* **1999**, *162*, 1–32. [[CrossRef](#)] [[PubMed](#)]
29. Athreya, K.B. Bootstrap of the Mean in the Infinite Variance Case. *Ann. Stat.* **1987**, *15*, 724–731. [[CrossRef](#)]
30. Arnold, T.; Emerson, J. Nonparametric goodness-of-fit tests for discrete null distributions. *R J.* **2011**, *3*, 34–39. [[CrossRef](#)]
31. Gillet-Markowska, A.; Louvel, G.; Fischer, G. *bz-rates*: A Web Tool to Estimate Mutation Rates from Fluctuation Analysis. *G3 Genes|Genomes|Genet.* **2015**, *5*, 2323–2327. [[CrossRef](#)]
32. Ramachandran, K.; Tsokos, C. *Mathematical Statistics with Applications*; Elsevier Science: Amsterdam, The Netherlands, 2009.
33. Zheng, Q. New algorithms for Luria-Delbrück fluctuation analysis. *Math. Biosci.* **2005**, *196*, 198–214. [[CrossRef](#)] [[PubMed](#)]
34. Feng, Z.; McCulloch, C.E. Statistical inference using maximum likelihood estimation and the generalized likelihood ratio when the true parameter is on the boundary of the parameter space. *Stat. Probab. Lett.* **1992**, *13*, 325–332. [[CrossRef](#)]
35. Ycart, B. *Modèles et Algorithmes Markoviens*; Springer Science & Business Media: Amsterdam, The Netherlands, 2002.
36. Mazoyer, A.; Drouilhet, R.; Despréaux, S.; Ycart, B. Flan: An R package for inference on mutation models. *R J.* **2017**, *9*, 334. [[CrossRef](#)]
37. Ycart, B.; Veziris, N. Unbiased estimation of mutation rates under fluctuating final counts. *PLoS ONE* **2014**, *9*, e101434; Erratum in *PLoS ONE* **2017**, *12*, e0173143. [[CrossRef](#)] [[PubMed](#)]
38. Ycart, B. Fluctuation analysis with cell deaths. *arXiv* **2012**, arXiv:1207.4375. Available online: <http://arxiv.org/abs/1207.4375> (accessed on 9 November 2023).
39. Ross, S. *Introduction to Probability Models*; Academic Press: Cambridge, MA, USA, 2007.
40. von Bortkewitsch, L. *Das Gesetz der Kleinen Zahlen*; B.G. Teubner: Leipzig, Germany, 1898.
41. Chiang, C.L. *Introduction to Stochastic Processes in Biostatistics*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1968.
42. Waller, J.H.; Rao, B.R.; Li, C.C. Heterogeneity of childless families. *Soc. Biol.* **1973**, *20*, 133–138. [[CrossRef](#)] [[PubMed](#)]
43. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2022.
44. Embrechts, P.; Hofert, M. A note on generalized inverses. *Math. Methods Oper. Res.* **2013**, *77*, 423–432. [[CrossRef](#)]
45. Crane, G. A modified Luria-Delbrück fluctuation assay for estimating and comparing mutation rates. *Mutat. Res. Mol. Mech. Mutagen.* **1996**, *354*, 171–182. [[CrossRef](#)]
46. de la Iglesia, F.; Martínez, F.; Hillung, J.; Cuevas, J.M.; Gerrish, P.J.; Daròs, J.A.; Elena, S.F. Luria-Delbrück Estimation of Turnip Mosaic Virus Mutation Rate In Vivo. *J. Virol.* **2012**, *86*, 3386–3388. [[CrossRef](#)]
47. Hall, B.M.; Ma, C.X.; Liang, P.; Singh, K.K. Fluctuation AnaLysis CalculatOR: A web tool for the determination of mutation rate using Luria-Delbrück fluctuation analysis. *Bioinformatics* **2009**, *25*, 1564–1565. [[CrossRef](#)]
48. Koch, A.L. Mutation and growth rates from Luria-Delbrück fluctuation tests. *Mutat. Res. Mol. Mech. Mutagen.* **1982**, *95*, 129–143. [[CrossRef](#)]
49. Kosterlitz, O.; Tirado, A.M.; Wate, C.; Elg, C.; Bozic, I.; Top, E.M.; Kerr, B. Estimating the rate of plasmid transfer with an adapted Luria-Delbrück fluctuation analysis. *bioRxiv* **2022**, bioRxiv:2021–01.

50. Lang, G.I. Measuring Mutation Rates Using the Luria-Delbrück Fluctuation Assay. In *Methods in Molecular Biology*; Springer: New York, NY, USA, 2017; pp. 21–31. [[CrossRef](#)]
51. Zheng, Q. A new practical guide to the Luria–Delbrück protocol. *Mutat. Res. Mol. Mech. Mutagen.* **2015**, *781*, 7–13. [[CrossRef](#)] [[PubMed](#)]
52. Zheng, Q. New approaches to mutation rate fold change in Luria–Delbrück fluctuation experiments. *Math. Biosci.* **2021**, *335*, 108572. [[CrossRef](#)]
53. Nakamura, M.; Pérez-Abreu, V. Empirical probability-generating function: An overview. *Insur. Math. Econ.* **1993**, *12*, 287–295. [[CrossRef](#)]
54. Alexander, H.K. Conditional Distributions and Waiting Times in Multitype Branching Processes. *Adv. Appl. Probab.* **2013**, *45*, 692–718. [[CrossRef](#)]
55. Lange, K. Calculation of the Equilibrium Distribution for a Deleterious Gene by the Finite Fourier Transform. *Biometrics* **1982**, *38*, 79–86. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.