OXFORD GENETICS

GENETICS, 2023, 225(3), iyad168
https://doi.org/10.1093/genetics/iyad168
Advance Access Publication Date: 19 September 2023
Investigation

Scaling the discrete-time Wright–Fisher model to biobank-scale datasets

Jeffrey P. Spence,[1,*] Tony Zeng,[1] Hakhamanesh Mostafavi,[1] Jonathan K. Pritchard[1,2]

[1]Department of Genetics, Stanford University, Stanford, CA 94305, USA
[2]Department of Biology, Stanford University, Stanford, CA 94305, USA

*Corresponding author: Department of Genetics, Stanford University School of Medicine, 291 Campus Drive, Mail Stop-5120, Stanford, CA 94305, USA.
E-mail: jspence@stanford.edu

The discrete-time Wright–Fisher (DTWF) model and its diffusion limit are central to population genetics. These models can describe the forward-in-time evolution of allele frequencies in a population resulting from genetic drift, mutation, and selection. Computing likelihoods under the diffusion process is feasible, but the diffusion approximation breaks down for large samples or in the presence of strong selection. Existing methods for computing likelihoods under the DTWF model do not scale to current exome sequencing sample sizes in the hundreds of thousands. Here, we present a scalable algorithm that approximates the DTWF model with provably bounded error. Our approach relies on two key observations about the DTWF model. The first is that transition probabilities under the model are approximately sparse. The second is that transition distributions for similar starting allele frequencies are extremely close as distributions. Together, these observations enable approximate matrix–vector multiplication in linear (as opposed to the usual quadratic) time. We prove similar properties for Hypergeometric distributions, enabling fast computation of likelihoods for subsamples of the population. We show theoretically and in practice that this approximation is highly accurate and can scale to population sizes in the tens of millions, paving the way for rigorous biobank-scale inference. Finally, we use our results to estimate the impact of larger samples on estimating selection coefficients for loss-of-function variants. We find that increasing sample sizes beyond existing large exome sequencing cohorts will provide essentially no additional information except for genes with the most extreme fitness effects.

Keywords: Wright–Fisher; selection; loss-of-function

Introduction

The discrete-time Wright–Fisher (DTWF) model and its large population limit the Wright–Fisher diffusion (WF diffusion) are workhorses of population genetics (Ewens 2004; Gillespie 2004). These forward-in-time models describe the evolution of the frequency of an allele in a population, and can incorporate mutation, selection, and genetic drift.

Beyond providing a useful conceptual framework, the DTWF model and the WF diffusion enable inference of evolutionary parameters from data. A notable example is the Poisson random field (PRF) model (Sawyer and Hartl 1992) which relates the distribution of allele frequencies at a single site to the probability of observing a given number of sites where an allele is at a particular frequency in the sample (the site frequency spectrum; SFS). The SFS can be estimated from sequencing data, and hence the PRF provides a probabilistic model relating evolutionary parameters to observable genetic data. Evolutionary parameters can then be inferred using standard techniques from statistical inference, such as maximum likelihood. This approach has been used to infer population sizes (Bhaskar *et al.* 2015), complex demographic models (Gutenkunst *et al.* 2009), and distributions of selection coefficients (Kim *et al.* 2017).

Unfortunately it is difficult to compute the distribution of frequencies at a single site for models with natural selection under either the DTWF model or the WF diffusion. This distribution is one of the key ingredients of the PRF model. To illustrate these difficulties, we will focus on the case of a single site with two potential alleles, A and a, in a panmictic monoecious haploid population.

Here, our overarching goal will be to compute the likelihood of observing a particular allele frequency at present given various evolutionary parameters such as past population sizes, mutation rates, and selection coefficients.

The simplest approach to computing these likelihoods is a naive forward-in-time application of the DTWF transition matrix. As we describe in more detail below, the crux of this naive method is repeatedly multiplying a vector of probabilities by a DTWF transition matrix, each of which require $O(N^2)$ time for a population of size N. Given that for humans, the present-day effective population size may be in the millions (or more) (Schiffels and Durbin 2014), this naive approach is obviously not scalable.

To avoid the onerous $O(N^2)$ runtime, many approaches are based on the idea that population sizes are usually quite large, and as such one might consider a large population size limit of the DTWF model. This limit assumes a fixed sample size n and then takes N to infinity, although in practice this approach is used for all but the smallest population sizes. The resulting continuum limit is the celebrated WF diffusion (Ewens 2004). The WF diffusion is still a Markov process, but instead of having a finite state space, the state space is the continuous unit interval [0, 1] of allele frequencies. As such, the limiting process is no longer a discrete time, discrete space Markov chain, but a continuous time, continuous space Markov process whose evolution is described

Received on 01 June 2023; accepted on 08 September 2023

by a stochastic differential equation. This stochastic differential equation then allows one to write a partial differential equation (PDE) that describes how the likelihood of observing different allele frequencies changes over time. Similar to the naive DTWF approach, one may solve this PDE at equilibrium at some time in the ancient past and then evolve the likelihoods forward in time to obtain the likelihoods at the present. The advantage of this approach is that whether the population has 10,000 individuals or 10 million individuals the PDE is functionally the same. As a result, the runtime of computing likelihoods becomes independent of N. This approach has been extremely fruitful, resulting in numerical solutions (Evans et al. 2007; Gutenkunst et al. 2009; Koch and Novembre 2017) and spectral approaches (Song and Steinrücken 2012; Steinrücken et al. 2013; Živković et al. 2015; Steinrücken et al. 2016). Yet, numerically solving a PDE can be difficult and error-prone, and some methods have been found to return negative "probabilities" (Kamm et al. 2017).

Another line of work takes a backward-in-time approach using ideas from Kingman's coalescent (Kingman 1982), resulting in likelihoods equivalent to those computed forward-in-time using the WF diffusion (Jansen and Kurt 2014) (but see Fu 2006; Bhaskar et al. 2015 for coalescent approaches that are equivalent to the DTWF model). These backward-in-time approaches have the advantage of only scaling in terms of the sample size, n, as opposed to the population size, N. Usually, $n \ll N$, and so this scaling can result in substantial computational speedups. For example, Polanski and Kimmel (2003) developed an approach to compute likelihoods under the coalescent for arbitrary past population size functions in $O(n^2)$ time. A major downside of coalescent approaches is the difficulty of incorporating natural selection (Krone and Neuhauser 1997). One can in principle obtain a genealogical process in the presence of natural selection (Krone and Neuhauser 1997), but inference under this process is generally intractable.

Under neutrality, results identical to the backward-in-time approach can be derived from a forward-in-time perspective by tracking the first n moments of the population frequency distribution, resulting in a system of n coupled ordinary differential equations (ODEs) (Evans et al. 2007). The approach in (Evans et al. 2007) can be extended to models with selection, but in that case, the system of ODEs is not "closed." This means that the time derivative of the nth moment of the population frequency distribution requires knowing all of the moments up to and including the $(n+2)$th moment for general models of selection, which in turn requires knowing the $(n+4)$th moment and so on.

Forward-in-time and backward-in-time methods each have advantages and disadvantages, and so a number of hybrid approaches have been developed. For example, `momi` (Kamm et al. 2017, 2020) uses the fact that the genealogies of the backward-in-time coalescent can be embedded in a forward-in-time Moran model. `momi` can model complex demographies, but cannot model selection. A similar trick is used in `moments` (Jouganous et al. 2017), but `moments` can model natural selection while remaining a good approximation to the WF diffusion.

While the WF diffusion and coalescent can enable more efficient inference, they are only accurate for sufficiently common alleles. This inaccuracy has been noted several times, usually in the context of the coalescent, but the coalescent and WF diffusion are dual processes, so these inaccuracies also apply to the WF diffusion. There begin to be notable discrepancies between the DTWF model and the WF diffusion when the sample size, n, is larger than roughly the square root of the population size, $\sqrt{N}$ (Wakeley and Takahashi 2003; Fu 2006; Bhaskar et al. 2014; Melfi and Viswanath 2018a; Krukov and Gravel 2021).

The diffusion limit also assumes that all relevant evolutionary parameters such as mutation rates and strengths of selection scale like $1/N$ in the limit (Ewens 2004). That is, if the selection coefficient is $\gg 1/N$, then the diffusion approximation breaks down (Krukov and Gravel 2021).

Most of the methods discussed above compute likelihoods under processes equivalent to the WF diffusion, potentially suffering from these problems. Bhaskar et al. (2014) introduced a coalescent approach dual to the DTWF model that scales like $O(n^3)$ but cannot incorporate natural selection. More recently, Krukov and Gravel (2021) developed an approach that can model natural selection using additional bookkeeping to accurately compute likelihoods under the DTWF process. Unfortunately, this approach scales like $O(n^4)$. The distinction between the DTWF process and the WF diffusion becomes apparent when n is larger than $O(\sqrt{N})$. In this regime, a runtime of $O(n^4)$ implies a runtime of at least $O(N^2)$, no better than the naive forward-in-time approach using the DTWF transition matrix.

There has been much interest in determining the extent to which natural selection acts against loss-of-function variants in each gene in the human genome using massive exome sequencing datasets (Lek et al. 2016; Cassa et al. 2017; Weghorn et al. 2019; Karczewski et al. 2020; LaPolice and Huang 2023; Agarwal et al. 2023). This regime—sample sizes in the hundreds of thousands, and extremely strong selection—is exactly where differences between the DTWF model and the diffusion become most pronounced, highlighting the need for more computationally efficient methods.

Here, we reconsider the naive approach of using the forward-in-time DTWF process. While the most basic method of computing likelihoods using the DTWF process costs $O(N^2)$ time, we show that transition matrices under a broad class of DTWF processes can be replaced by highly structured matrices enabling likelihood computations in $O(N)$ time, while having a provably small approximation error. We obtain a similar speedup for Hypergeometric sampling that may be of independent interest. We provide a high level description of our method in the *Overview of approach* section, and we show empirically that our approach is highly accurate and can scale to sample sizes in the tens of millions in the *Runtime and accuracy* section. We use our approach to explore the utility of using loss-of-function variants to estimate selection coefficients in large samples in the *Impact of mutation, selection, and demography on the DTWF model* section. We find that increasing sample sizes beyond current values will provide little value for estimating the selection coefficients of most genes, and will only prove useful for estimating extremely strong selection coefficients. We discuss the limitations and future directions for our approach in the *Discussion* section. Formal proofs are deferred to Appendix A. We apply our approach in an empirical Bayes framework to estimate the strength of selection against loss-of-function variants using large-scale exome sequencing data in a companion paper (Zeng et al. 2023). Software with a python API implementing our approach is available at https://github.com/jeffspence/fastDTWF.

Overview of approach

Throughout, we focus on a single locus with two alleles, A and a. Our goal is to compute the likelihood of observing a given frequency of the A allele at a particular locus in a sample from a population evolving according to the DTWF model. We will use the notation $\mathbf{v}_t$ to represent a vector of these likelihoods at generation t. That is, entry i of $\mathbf{v}_t$ is the likelihood of observing exactly i copies of the A allele in the population at generation t. Thus, if we say that the present is generation T, our goal is to compute $\mathbf{v}_T$. All of the evolutionary forces present in a general DTWF model are captured by the transition matrices, $\mathbf{M}_t$. In particular, $\mathbf{M}_t$ is affected by the population sizes in generations t and $t+1$, as well as the mutation rates and effects of selection. For general nonequilibrium populations where these evolutionary parameters are changing over time, a naive approach to computing these likelihoods involves three steps:

1) We assume that at some point in the past the population was at equilibrium, and we compute $\mathbf{v}_0$, a vector with $N+1$ entries, indexed from 0 to N, where entry i is the probability of observing i copies of the A allele in the population at equilibrium, with N being the population size.
2) We evolve these probabilities forward according to the DTWF transition matrix for each generation, until we reach the present. That is, for generation $t-1$, let $\mathbf{M}_{t-1}$ be the DTWF transition matrix. Then, $(\mathbf{M}_{t-1})_{i,j}$ is the probability of going from i copies of the A allele in the population in generation $t-1$ to j copies of the A allele in generation t. To obtain the probability of each allele frequency in the population at generation t, we can compute

$$\mathbf{v}_t = \mathbf{M}_{t-1}^{\mathsf{T}} \mathbf{v}_{t-1},$$

where we use the T superscript to denote matrix transposition. Say that we want to compute the likelihood at generation T. Then, given $\mathbf{v}_0$, we can compute the population-level allele frequency probabilities as

$$\mathbf{v}_T = \mathbf{M}_{T-1}^{\mathsf{T}} \cdots \mathbf{M}_0^{\mathsf{T}} \mathbf{v}_0.$$

3) The first two steps compute the probability of observing each different possible allele frequency in the *population*, and so we still must obtain the probability of observing each different possible allele frequency in a *sample* from the population. Let $\mathbf{S}$ be a matrix where entry $\mathbf{S}_{i,j}$ is the probability of seeing j A alleles in a sample given that there are i A alleles in the population. We may therefore obtain the probabilities of observing each possible allele frequency in the sample, $\mathbf{v}_{\text{sample}}$ as

$$\mathbf{v}_{\text{sample}} = \mathbf{S}^{\mathsf{T}} \mathbf{v}_T.$$

Each of these steps is intractable using a naive approach because they rely on matrix–vector multiplication, requiring $O(N^2)$ time. Yet, if we were able to make matrix–vector multiplication much faster, then this approach suddenly becomes attractive—it is conceptually straightforward; easy to extend to incorporate selection and changes in mutation rates or population sizes; and numerically stable because all of the entries in all of the vectors and matrices are positive, avoiding the catastrophic cancellation that plagues some other approaches.

Our approach is to replace the DTWF transition matrices $\mathbf{M}_t$ and the sampling matrix $\mathbf{S}$ with approximate versions, $\widetilde{\mathbf{M}}_t$ and $\widetilde{\mathbf{S}}$ that allow for matrix–vector multiplication in $O(N)$ time, while guaranteeing that $\widetilde{\mathbf{M}}_t$ and $\widetilde{\mathbf{S}}$ are extremely "close" to $\mathbf{M}_t$ and $\mathbf{S}$, respectively.

Our main results are about matrices where each row consists of the $N+1$ entries of a probability mass function of a Binomial distribution with sample size N. We call this class of matrices Binomial transition matrices. While this class of matrices may seem esoteric, the transition matrices of many types of DTWF models are either themselves Binomial transition matrices or can be well approximated using Binomial transition matrices. On an intuitive level, our results rely on the observation that all of the "action" in a Binomial distribution happens on the scale of $O(\sqrt{N})$ in a way that we describe in more detail below.

To obtain sampling probabilities, we generally consider sampling without replacement from the population. As we will describe below, this results in $\mathbf{S}$ being a matrix where each row is the probability mass function of a Hypergeometric distribution. The same properties of Binomial distributions that allow us to perform fast matrix–vector products are also true of Hypergeometric distributions. This allows us to use very similar tricks to quickly compute matrix–vector products with the sampling matrix, $\mathbf{S}$.

In contrast, if one obtains a sample via sampling *with* replacement, then sampling can be represented as one additional generation of a DTWF process, but with no mutation and no selection. One can also model sampling with replacement where the sampling is biased toward individuals with one allele or the other, in which case the sampling process is equivalent to a single generation of the DTWF model without mutation, but with a particular form of natural selection. Such a situation might arise when sampling case/control data, where cases are over-represented relative to their abundance in the population. In this case, there would be a bias toward sampling disease-associated alleles. More complicated sampling processes (e.g. sampling without replacement, but biased toward one allele or another) may be possible to treat using our techniques, but would require additional considerations beyond those presented here.

Total variation distance and matrix norms

Our results about Binomial and Hypergeometric distributions are in terms of total variation distance, an important metric on the space of distributions. See Gibbs and Su (2002) for a comprehensive overview of metrics on the space of distributions, including total variation distance. For the discrete distributions taking values in 0, 1, 2, ..., N that we consider here, total variation distance is simply half the ℓ_1 distance between the probability mass functions. That is, for a distribution P and a distribution Q, we write the total variation distance between them, $d_{\text{TV}}(P, Q)$, as

$$d_{\text{TV}}(P, Q) := \frac{1}{2} \sum_{k=0}^{N} \left| \mathbb{P}\{X = k\} - \mathbb{P}\{Y = k\} \right|,$$

where X is a P-distributed random variable and Y is a Q-distributed random variable.

We present our results in terms of total variation distance because it is very closely related to a particular matrix norm. The 1-*operator norm* of a matrix, which we denote by $\|\cdot\|_1$, is defined as

$$\|A\|_1 := \sup_{x : \|x\|_1} \|Ax\|_1$$

for a matrix A, where $\|\cdot\|_1$ applied to vectors is the usual ℓ_1 norm (i.e. the sum of the absolute values of the entries). Note that the 1-operator norm is *not* the sum of the absolute values of the entries of the matrix. In fact, the 1-operator norm is the maximum of the column-wise ℓ_1 norms. The proof of this well-known result is included in Appendix B for completeness.

The reason we are interested in the 1-operator norm is because it allows us to bound how much error we might introduce by replacing a DTWF transition matrix by an approximation. In particular, if we have a DTWF transition matrix $\mathbf{M}$ and we can construct an approximate matrix $\widetilde{\mathbf{M}}$ such that $\|\mathbf{M}^{\mathsf{T}} - \widetilde{\mathbf{M}}^{\mathsf{T}}\|_1 \le \varepsilon$, then when we compute the matrix–vector products required for computing likelihoods, we will have that $\|\mathbf{M}^{\mathsf{T}}\mathbf{v} - \widetilde{\mathbf{M}}^{\mathsf{T}}\mathbf{v}\|_1 \le \varepsilon$. Since the rows of a DTWF matrix correspond to probability mass functions, we see that bounding an approximation's row-wise ℓ_1 distance is equivalent to bounding the total variation distance between the corresponding distributions up to a factor of 2.

Binomial transition matrices are approximately sparse

The first key for our approach is straightforward: Binomial random variables are very unlikely to be too far away from their means. A Binomial random variable will be more than

$$\sqrt{\frac{N}{2}\log\frac{2}{\varepsilon}}$$

away from its mean with probability less than ε. This is a celebrated result due to Hoeffding (1963), and shows that with overwhelming probability, a Binomial random variable will be within a constant factor times $\sqrt{N}$ of its mean. In turn, this implies that we can ignore all but $O(\sqrt{N})$ entries in each row of a Binomial transition matrix while only incurring a constant (in N) total variation distance. This property of Binomial distributions is illustrated in Fig. 1a.

This simple observation alone provides substantial savings in terms of memory and runtime for computing matrix–vector products, which has been noted previously (Krukov et al. 2016; Tataru et al. 2016), as we can simply take each row of a Binomial transition matrix, and replace all of the entries that are too far away from the corresponding mean of the row by zero. We can choose the point at which we begin setting entries to zero to obtain a given error tolerance, ε. To ensure that the resulting approximate matrix is still a valid stochastic matrix (i.e. the rows still sum to one and hence are valid probability distributions), we divide the remaining nonzero entries by their sum, which by construction perturbs them by a multiplicative factor no larger than $1/(1-\varepsilon)$.

As a concrete example, to capture all but 10^{-16} of the probability in a Binomial distribution, corresponding roughly to the limits of numerical precision, we can ignore all but $\approx 4.33\sqrt{N}$ entries on either side of the mean when computing matrix–vector products. This means that we can approximate the matrix as having only $\approx 8.66\sqrt{N}$ nonzero entries in each row, resulting in a theoretical speedup by a factor of $\sqrt{N}/8.66$. When N is 1,000, this corresponds to a factor of ≈7.3× speedup over the naive approach. When N is 10 million, the speedup is ≈730×. This high degree of sparsity is visually apparent in the 5, 000 × 5, 000 neutral DTWF transition matrix shown in Fig. 1b.

Binomial transition matrices are approximately low rank

The second key for our approach is more subtle, and relies on the fact that Binomial distributions with similar success probabilities have similar distributions in terms of total variation distance. This makes sense on an intuitive level—flipping N coins that come up heads with probability p should result in a similar distribution of outcomes to flipping N coins that come up heads with probability $p + \delta$. What is less obvious is the length scale at which this occurs. That is, how large can δ be in terms of p and N while still keeping the total variation distance between the two distributions below a specified level, ε? The answer is $c_\varepsilon\sqrt{p(1-p)/N}$ for some constant c_ε that depends on ε but is independent of p and N. This scaling is visually apparent in the different colored points in Fig. 1a, and we prove this result in Appendix A.

We now consider partitioning the unit interval [0, 1] into blocks such that for p and p' in the same block, two Binomial distributions with size N and success probabilities p and p' will have total variation distance less than ε. We show in "Formal theoretical results and proofs" that we can achieve such a partitioning of [0, 1] with $O(\sqrt{N})$ blocks.

We can use this partitioning to approximate a Binomial transition matrix by an extremely low rank matrix, while bounding the ℓ_1 error introduced to each row. For each separate block in the partition of [0, 1], we can pick a representative success probability. Since there are $O(\sqrt{N})$ blocks in this partition, we end up with $O(\sqrt{N})$ representative success probabilities. Then, for each row of the Binomial transition matrix, we can consider its success probability, determine which block of the partition it is in, and replace that row by the probability mass function of the Binomial random variable with the corresponding representative success probability. Because the original row and the new row correspond to probability mass functions for Binomial distributions that are close in total variation distance, the rows are close in ℓ_1 distance. After replacing each row, the resulting matrix can have at most $O(\sqrt{N})$ unique rows, meaning that for large N it is extremely low rank. Additionally, since each row of the resulting matrix is still the probability mass function of some Binomial distribution, the resulting matrix is still a Binomial transition matrix.

While picking an arbitrary success probability for each block in the partition of [0, 1] bounds the total variation distance, different choices can results in different accuracies of the approximation over repeated matrix–vector multiplications. For instance, if one were to choose the smallest success probability within each block, then, for a neutral DTWF transition matrix, the expected frequency in the next generation would never be larger than the current frequency and would often be slightly smaller. Over evolutionary time-scales this would act similarly to negative selection, affecting the long-term accuracy. Instead of choosing arbitrarily, we found that in practice a moment-matching approach is extremely accurate. Briefly, when performing matrix–vector multiplication with a nonnegative vector $\mathbf{v}$, for a given block of the partition of [0, 1] we use the weighted average of the success probabilities in $\mathbf{M}$ that fall within that block with weights proportional to the corresponding entries of $\mathbf{v}$.

Specifically, suppose that rows $i, i+1, \ldots, j$, with success probabilities $p_i, p_{i+1}, \ldots, p_j$ will all be represented by a single row with success probability $\widetilde{p}_k$. If we assume that the success probabilities are ordered, then setting $\widetilde{p}_k$ to be any value between p_i and p_j will bound the total variation distance, but some choices may result in worse long-term accuracy. If we are approximating $\mathbf{M}^{\mathsf{T}}\mathbf{v}$ for some $\mathbf{v}$ with nonnegative entries, then we set

$$\widetilde{p}_k = \frac{\sum_{\ell=i}^{j} p_\ell \mathbf{v}_\ell}{\sum_{\ell=i}^{j} \mathbf{v}_\ell}.$$

This choice guarantees that the expected frequency in the next generation of an allele with frequency in the current generation chosen with probabilities proportional to $\mathbf{v}$ is matched between the true and approximate processes. In the event that the denominator is exactly 0, then the choice of $\widetilde{p}_k$ does not matter as we will see in the next section.

An $O(N)$ algorithm for approximately computing matrix–vector products for binomial transition matrices

These two ingredients—that each row of a Binomial transition matrix is close in ℓ_1 distance to a row with only $O(\sqrt{N})$ nonzero

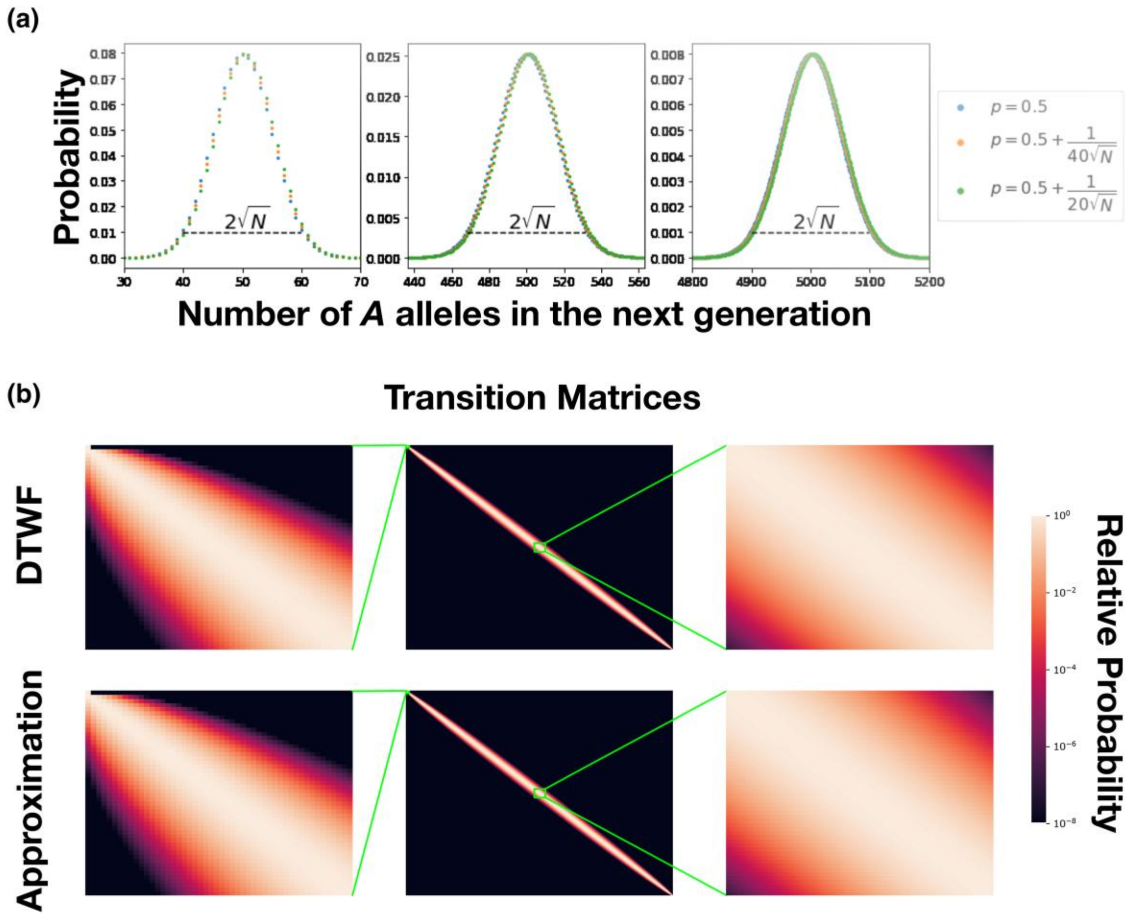

Fig. 1. a) Probability mass functions for Binomial distributions across a range of values of N. Most of the mass is contained within $O(\sqrt{N})$ of the mean, and distributions with success probabilities p within a small factor of $1/\sqrt{N}$ of each other are virtually indistinguishable. b) The transition matrix of the neutral DTWF process with $N = 5,000$ as well as our approximation of that matrix represented as a heatmap. Rows are normalized so that the maximum of each row is 1, and regions from the top left and middle are expanded. The results are nearly indistinguishable, except that there is very subtle horizontal banding near the middle of the transition matrix resulting from having nearby rows be copies of each other.

entries, and that each row of a Binomial transition matrix is close in ℓ_1 distance to the corresponding row of a Binomial transition matrix with only $O(\sqrt{N})$ unique rows—are sufficient to derive a substantially faster algorithm for approximately computing (transposed) matrix–vector products.

The key idea is to replace the original Binomial transition matrix $\mathbf{M}$ by an approximation $\widetilde{\mathbf{M}}$, which we construct by first choosing a Binomial transition matrix with $O(\sqrt{N})$ unique rows that is close in row-wise ℓ_1 distance to $\mathbf{M}$ and then sparsifying each row of that matrix. If we perform each of these steps so that they introduce a row-wise ℓ_1 distance of at most $\varepsilon/2$, the triangle inequality implies that the two steps together introduce a total row-wise ℓ_1 error of at most ε per row.

Once we have our approximate matrix, we can quickly perform matrix–vector multiplications (Fig. 2). The algorithm involves noting that computing $\widetilde{\mathbf{M}}^{\mathsf{T}}\mathbf{v}$ can be thought of as first multiplying each row of $\widetilde{\mathbf{M}}$ by the corresponding entry of $\mathbf{v}$, and then summing up those resulting vectors. That is, one first performs $N + 1$ scalar–vector multiplications, and then sums up the $N + 1$ resulting vectors. Our speedups come from two places.

First, instead of multiplying each identical row by the corresponding entry of $\mathbf{v}$ and then summing, we can instead first sum up all of the entries of $\mathbf{v}$ that correspond to identical rows, and then multiply one representative row by this sum of the relevant entries of $\mathbf{v}$. This observation means that after grouping the entries of $\mathbf{v}$, we only need to perform $O(\sqrt{N})$ scalar–vector multiplications, and then sum up the $O(\sqrt{N})$ resulting vectors.

Second, since each row of $\widetilde{\mathbf{M}}$ is sparse, we can ignore all of the zero entries when performing scalar–vector multiplication and vector addition. Our vectors only have $O(\sqrt{N})$ nonzero entries, making both of those operations cost $O(\sqrt{N})$ time. Overall, this means that we must perform $O(\sqrt{N})$ operations, each taking $O(\sqrt{N})$ time, resulting in a runtime of $O(N)$. We give a visual depiction of our algorithm in Fig. 2. Details and technical proofs are presented in Appendix A.

There are a few technical details and assumptions in achieving a truly $O(N)$ runtime. First, we obviously cannot store or even look at each entry in $\mathbf{M}$, because doing so would require $O(N^2)$ space and time. Instead we represent a Binomial transition matrix (or DTWF matrix) as simply the $N + 1$ success probabilities corresponding to each row. We can then represent $\widetilde{\mathbf{M}}$ by storing only the locations and values of the $O(\sqrt{N})$ entries for each of the $O(\sqrt{N})$ unique rows. Second, determining which representative row to use for each row of $\mathbf{M}$ can—in the worst case—require $O(N \log N)$ time; one can keep the break points of the partition of [0, 1] in an ordered list, and then for each row of $\mathbf{M}$ one must take its corresponding success probability and search through the sorted list of break points. This binary search requires $O(\log N)$ time for each row, resulting in a runtime of $O(N \log N)$. Instead, we assume that the rows of $\mathbf{M}$ are ordered in terms of success probabilities. In Appendix A, we present a simple $O(N)$

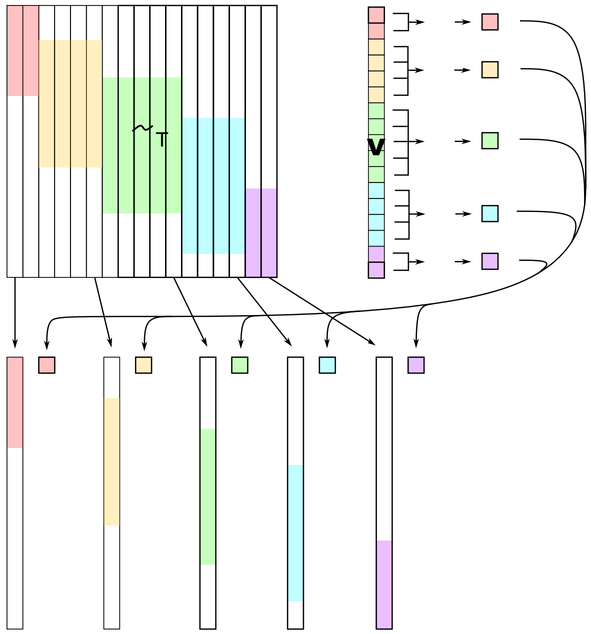

Fig. 2. Schematic of fast matrix–vector multiplication algorithm. Blank regions of $\tilde{\mathbf{M}}^{\mathsf{T}}$ correspond to zeros. The algorithm proceeds by first combining entries of $\mathbf{v}$ that correspond to identical rows of $\mathbf{m}$, then multiplying the resulting scalars by the representative rows of $\tilde{\mathbf{M}}$, using sparse scalar–vector multiplication. The resulting sparse vectors are then summed using sparse vector addition.

algorithm for assigning ordered success probabilities to blocks of the partition. In the DTWF context, this ordering corresponds to the case where the expected allele frequency in the next generation is nondecreasing in the current allele frequency. This assumption is biologically reasonable, and would only be violated by something like extreme and unusual density-dependent selection. Indeed, this assumption holds for standard models of haploid or diploid selection and mutation.

Efficiently obtaining the likelihood of a sample from population likelihoods

Using the algorithm in the previous section allows us to compute population-level likelihoods. In general, we do not have access to population-level data and must instead obtain a sample from the population, which we assume is done uniformly at random without replacement (i.e. simple random sampling). If we take a sample of size n, then supposing that there are K copies of the A allele in the population, the number of A alleles in the sample is Hypergeometric distributed with parameters $N, K,$ and n. Thus, to obtain the probability of observing a given number of A alleles in the sample, we must take an average of Hypergeometric distributions weighted by the probability of having a given number of A alleles in the population. We can write this as a matrix equation, using an $(N+1)\times(n+1)$ dimensional sampling matrix $\mathbf{S}$ whose Kth row is the probability mass function of a Hypergeometric random variable with parameters $N, K,$ and n (assuming that the rows are 0-indexed). If the probabilities of observing $0, 1, \ldots, N$ copies of the A allele are contained in the $N+1$ dimensional vector $\mathbf{v}$, then we can obtain the vector, $\mathbf{v}_{\text{sample}}$ of probabilities of observing $0, 1, \ldots, n$ copies of the A allele as

$$\mathbf{v}_{\text{sample}} = \mathbf{S}^{\mathsf{T}}\mathbf{v}.$$

Naively computing this matrix–vector product would take $O(nN)$ time and space, which is prohibitive for large sample sizes.

Somewhat surprisingly, if we assume that the sample size is not too large as a function of the population size—i.e. $n \le \alpha N$ for any fixed $\alpha < 1$ as N grows, then the matrix $\mathbf{S}$ has properties very similar to a Binomial transition matrix despite having Hypergeometric rows instead of Binomial rows. On the one hand, the difference between the Hypergeometric distribution and the Binomial distribution is the same as sampling with and without replacement. Thus, for samples that are small, we might expect that it would be rare to sample the same individual twice when sampling with replacement, and so sampling with and without replacement should be similar. On the other hand, our results apply even as the sample size grows with the population size, and even for cases where, for example, we are sampling 99% of the population. In such cases, when sampling the same number of individuals, but *with* replacement we would almost certainly resample some individuals multiple times, and so it is surprising that sampling with and without replacement might have similar properties in this regime. Yet, as we show in Appendix A, $\mathbf{S}^{\mathsf{T}}$ is close in 1-operator norm to a highly structured matrix $\tilde{\mathbf{S}}^{\mathsf{T}}$ such that $\tilde{\mathbf{S}}$ has $O(\sqrt{n})$ unique rows and each row of $\tilde{\mathbf{S}}$ has $O(\sqrt{n})$ nonzero entries. The proof of this result relies on similar considerations as the Binomial case but applied to Hypergeometric distributions. Together, these two properties were all that was required to obtain our $O(N)$ algorithm for (transposed) matrix–vector products, and so we may use exactly the same trick to compute $\mathbf{S}^{\mathsf{T}}\mathbf{v}$.

As above, some care needs to be taken when choosing a row as a "representative" of a set of similar rows. In this case, we again use a moment-matching approach, taking a mixture of two Hypergeometric distributions so that the expected value of an allele chosen uniformly at random with probabilities proportional to $\mathbf{v}$ is matched between the approximate and real sampling matrices.

This completes our overview of our approach to approximately computing likelihoods. By applying our algorithm to the DTWF transition matrix, we can efficiently compute the stationary distribution, and integrate that distribution forward to the present, obtaining the present-day population-level likelihood. Then, by applying essentially the same algorithm to the sampling matrix, we can efficiently obtain sample-level likelihoods.

Numerical results

In this section, we present numerical results about the runtime and accuracy of our method as well as an application to how selection and demography interact to affect the distribution of observed frequencies in large sample sizes. These results have implications for how much information we might hope to gain about selection coefficients as sample sizes grow.

Before presenting the numerical results, we discuss some practical implementation details. The theoretical accuracy guarantees of our approach involve implicit constants. For example, we know that we can choose a c_ε such that if we replace a row of the transition matrix that corresponds to a Binomial with success probability p, with a row that corresponds to a success probability of $p + c_\varepsilon\sqrt{p(1-p)/N}$, then we induce an ℓ_1 error that is bounded by ε regardless of N or p. Yet, our proof that such a constant exists is nonconstructive (and our proof is such that making it constructive would result in a much smaller c_ε than necessary). As a result, we instead have the user specify two hyperparameters, which together determine both the runtime and accuracy. Regardless of their setting, our matrix–vector multiplication runs in $O(N)$ time, but the hyperparameters determine the size of the constant hidden by the big-O notation, as well as the accuracy. The first hyperparameter is the c_ε described above, which can alternatively be described as how many standard deviations away two rows' success probabilities can be before we will not allow one to be a copy of the other. Setting this value to be smaller results in a longer runtime, but higher accuracy. Unless otherwise specified, we set c_ε to be 0.1 for all analyses. The other hyperparameter, which we will denote by $\varepsilon_{\text{sparse}}$, determines how sparse to make the rows. Here, our theoretical results *do* provide a constructive guarantee, so the user specifies a particular row-wise ℓ_1 error tolerance, and the rows are only sparsified to a level that guarantees a smaller error. Again, setting $\varepsilon_{\text{sparse}}$ to be smaller results in longer runtimes, but higher accuracy. Unless otherwise specified, we set this $\varepsilon_{\text{sparse}}$ to be 10^{-8}. We also use the same hyperparameters to specify the level of accuracy for the sampling matrix, $\mathbf{S}$, but allow the user to set them to different values. Throughout, we always set c_ε for the sampling matrix to 0.05 and $\varepsilon_{\text{sparse}}$ for the sampling matrix to 10^{-8}.

Runtime and accuracy

To begin, we confirm the linear runtime of our matrix–vector multiplication algorithm and compare our implementation to the runtime of the naive quadratic approach. Throughout this section, we consider the neutral case with bidirectional recurrent mutation, where we have that the success probability, $p(f)$, for a given allele frequency in the parental generation, f, is

$$p(f) := \mu_{0\to1}(1-f) + (1-\mu_{1\to0})f,$$

where $\mu_{i\to j}$ is the probability of mutating from allele i to allele j. For the analyses in this section, we take $\mu_{0\to1} = \mu_{1\to0} = 1.25\times10^{-8}$.

We compared the runtime of matrix–vector multiplication with the DTWF transition matrix for this process and a random vector, where each entry is independent and identically distributed Uniform(0, 1), and then normalized to sum to one. Compared to the $O(N^2)$ naive matrix–vector multiplication algorithm, our approximate algorithm has a more favorable $O(N)$ scaling (Fig. 3a). While big-O notation hides constant factors, we see that across population sizes (from $N \geq 1,000$) our approximate algorithm is faster than the naive algorithm. Indeed, at $N = 1,000$, our approximate algorithm is slightly faster (6% speedup), while at $N = 10{,}000{,}000$, we predict that our algorithm would be about 215,000× faster, with the naive algorithm predicted to take over 75 days and our algorithm taking 31 s (we did not run the naive algorithm for $N > 79{,}000$ due to the prohibitive runtime). Similar results hold for matrix–vector multiplication using the sampling matrix (Fig. 3b), where we obtain substantial speedups regardless of whether we sample 5% or 50% of the population.

Having confirmed that our approximate matrix–vector multiplication algorithm provides a substantial speedup, we turned to assessing its accuracy. We began by considering the accuracy of performing a single matrix–vector multiplication. That is, we considered the ℓ_1 error between the result of the exact matrix–vector multiplication algorithm, and the approximate matrix–vector multiplication algorithm, $\|\mathbf{M}^\top\mathbf{v} - \widetilde{\mathbf{M}}^\top\mathbf{v}\|_1$. We used the same randomly generated $\mathbf{v}$ as we used for benchmarking the runtimes. Our theoretical results imply that even as N grows, our approximation should not get any worse. This theory is borne out empirically, where in fact we see that the error is small across all N, and the approximation actually appears to become more accurate as N gets large (Fig. 3c).

We see similar results for the sampling matrix (Fig. 3d), where the error is slightly larger when sampling a larger fraction of the population, but is low across all N and both of the sampling fractions considered. When the population size is small and we sample 50% of it, we see almost zero error. The reason for this is subtle, but by our construction of the approximate sampling matrix, $\widetilde{\mathbf{S}}$ if no more than two rows are combined into any single representative row, then our algorithm exactly recapitulates matrix–vector multiplication (up to the error induced by sparsification). For these examples, we chose the sparsification parameter, $\varepsilon_{\text{sparse}}$, to bound the ℓ_1 error by 10^{-8}, and so in this regime, the error we see is $\leq 10^{-8}$. As the population size increases, we begin combining more than two rows into any single representative row, and consequently we incur errors on the order of 10^{-5}.

The explanation for the increasing accuracy as N gets large is that our theory applies for *any* $\mathbf{v}$, and as such is a "worst-case" bound. In contrast, our benchmarks use random $\mathbf{v}$, and thus approximate an average case. Intuitively, our approximation results in the mean of the distribution corresponding to each row of the transition matrix being off by a little bit, and our theory bounds how off any single row can be. Yet, the mean of the distribution corresponding to some rows will be slightly too large and for others it will be slightly too small. If $\mathbf{v}$ has similar entries for a row whose mean is too large and a row whose mean is too small, some of the resulting errors will cancel. When $\mathbf{v}$ is random, as N gets large, more rows get grouped into a single representative row, and so there are more chances for the rows with means that are too large and too small to cancel each other out. As we will show below, in realistic scenarios (i.e. computing likelihoods) our algorithm is actually even more accurate than suggested by Fig. 3. This is because likelihoods tend to be very smooth across frequencies, resulting in cancellation of errors.

Our theoretical guarantees only hold for a single matrix–vector multiplication. In theory, an approximation that is very good for a single step can become essentially arbitrarily bad over multiple rounds of matrix–vector multiplication. As such, we numerically explored the long-term accuracy of our approximation by computing transition mass functions (TMFs)—the probability of observing a given allele frequency at a particular point in the future given some current frequency. The TMF can be computed by repeatedly multiplying a vector with all zeros except for a one at the entry corresponding to the present-day frequency with the single generation transition matrix. Our theory guarantees that our fast matrix–vector multiplication algorithm will result in a highly accurate approximation to the true TMF for a small number of generations, but our theory cannot determine whether the approximation gets worse over time.

To explore this, we considered a neutral model with bidirectional mutation at a rate of 1.25×10^{-8}, and a population size of 2,000, and computed TMFs for an initial frequency of 10%. To assess the accuracy of our approximation, we considered both the total variation distance and the symmetrized Kullback–Leibler (KL) divergence between the approximate and true TMFs. The KL divergence between two probability mass functions p and q is

$$D(p\|q) := \sum_x p(x)\log\frac{p(x)}{q(x)}.$$

This divergence is asymmetric, so we consider the symmetrized version $D(p\|q) + D(q\|p)$. The KL divergence is zero if and only if the two mass functions are identical, and small values indicate that the distributions are "close" in an information theoretic sense. Very roughly speaking, $1/D(p\|q)$ is approximately the

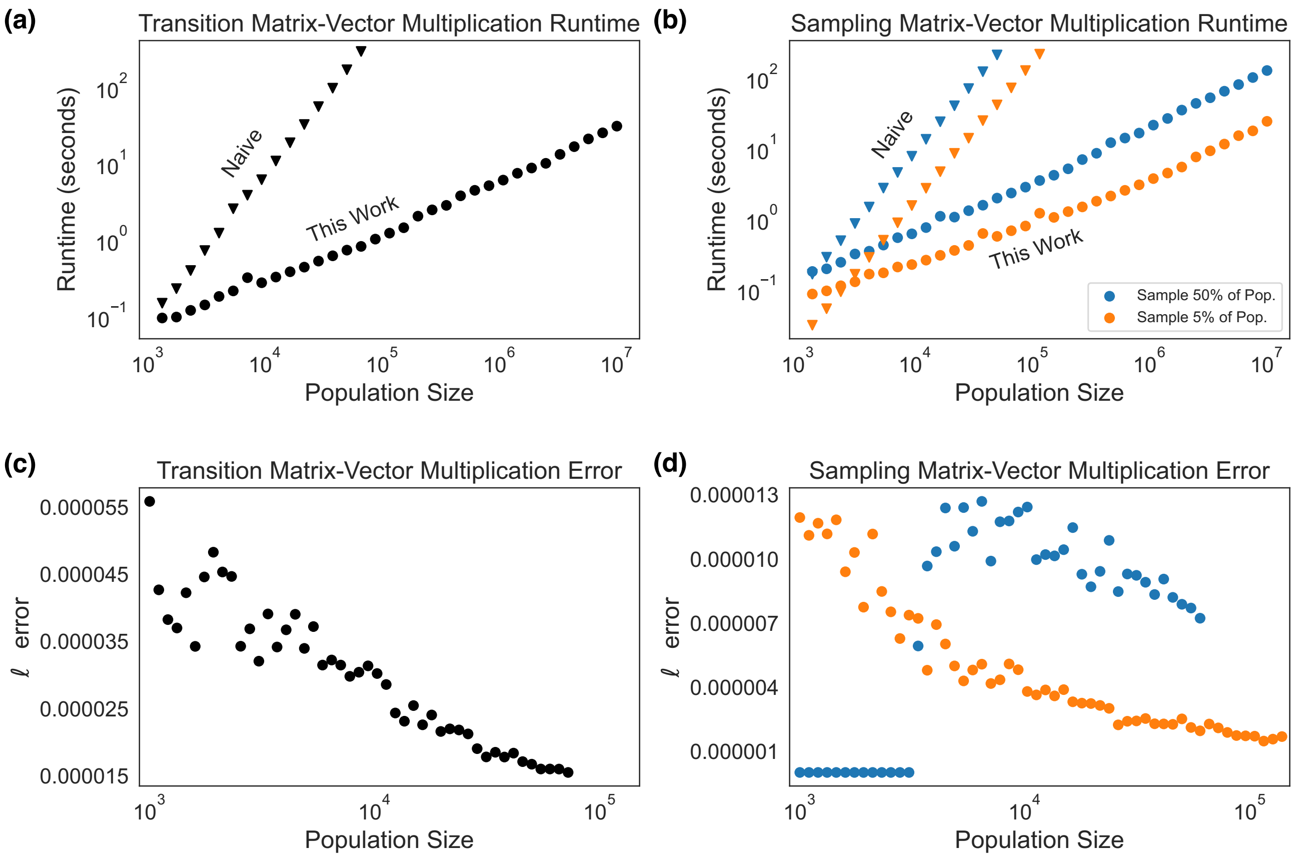

Fig. 3. Runtime and accuracy of approximate algorithm. a) Runtime of the approximate and naive matrix–vector multiplication algorithms for a DTWF transition matrix as a function of the population size, plotted on log–log scale. The naive algorithm scales quadratically, while the approximate algorithm proposed here scales linearly. b) Runtime of the approximate and naive matrix–vector multiplication for the sampling matrix as a function of the population size, plotted on log–log scale. Runtimes are shown for sample sizes that are 5% or 50% of the population size. In both a) and b), runs that were expected to take more than 5 min were not run. c) The ℓ_1 error of the vector resulting from our approximate matrix–vector multiplication algorithm, compared to the vector obtained from exact matrix–vector multiplication for a DTWF transition matrix multiplied with a random vector. d) Same as c), but for the sampling matrix, when considering sampling either 50% (blue) or 5% (orange) of the population. In both c) and d) it is apparent that the ℓ_1 error does not grow (and in fact decreases) with increasing population size, consistent with our theoretical guarantees.

number of independent observations one would need in order to distinguish the two distributions. The results are presented in Fig. 4a, where we show that even at long time-scales, our approximation remains extremely accurate. We show example TMFs for 10, 100, and 1, 000 generations in the future, where the approximation is visually indistinguishable from the true TMF (Fig. 4b).

Going even further, we can consider whether we are able to recover the long-term equilibrium of the DTWF process using our approximation. To investigate this, we computed SFSs under the DTWF model. We define the SFS more precisely in Appendix C but briefly, the SFS arises in the infinite sites mutation model, which approximates the case where each position in the genome has a very low mutation rate (and hence is very unlikely to be segregating) but there are many positions across the genome, so one still expects to find some segregating sites. In this regime, mutations can only happen once per site, and so it is possible to distinguish the ancestral allele from the derived allele. The SFS is then a vector where the ith entry is the number of positions in the genome at which there are i derived alleles, where i ranges between 1 and $n - 1$, inclusive, with n being the sample size. Here, we consider the normalized SFS, where this vector is normalized to sum to 1, which can be interpreted as the distribution over the number of derived alleles at a randomly chosen segregating site. The normalized SFS is commonly used for demographic inference and the inference of selection coefficients (Gutenkunst et al. 2009; Kim et al. 2017; Spence and Song 2019).

Computing the SFS requires finding the equilibrium of a particular system, and hence can be thought of as the limit of taking infinitely many matrix–vector products. As a result, our theory on the accuracy of our approximation does not apply, and so we explored the accuracy numerically.

To compute equilibria using our approximation, we view all of the frequencies corresponding to a given representative success probability as a single "meta-state." We then build the Markov transition matrix on these meta-states implied by our approximation to the DTWF process. Finding the equilibrium of the Markov chain on the meta-states involves solving a matrix equation with the $O(\sqrt{N}) \times O(\sqrt{N})$ transition matrix. The resulting meta-state equilibrium is then converted to an equilibrium in the original state space by multiplying the amount of mass in each meta-state by the (truncated) Binomial PMF with that meta-state's corresponding representative success probability. See Appendix C for more details.

We compared the accuracy of our approximation to the commonly used diffusion approximation (Gutenkunst et al. 2009; Bhaskar et al. 2015; Jouganous et al. 2017; Kamm et al. 2017), with

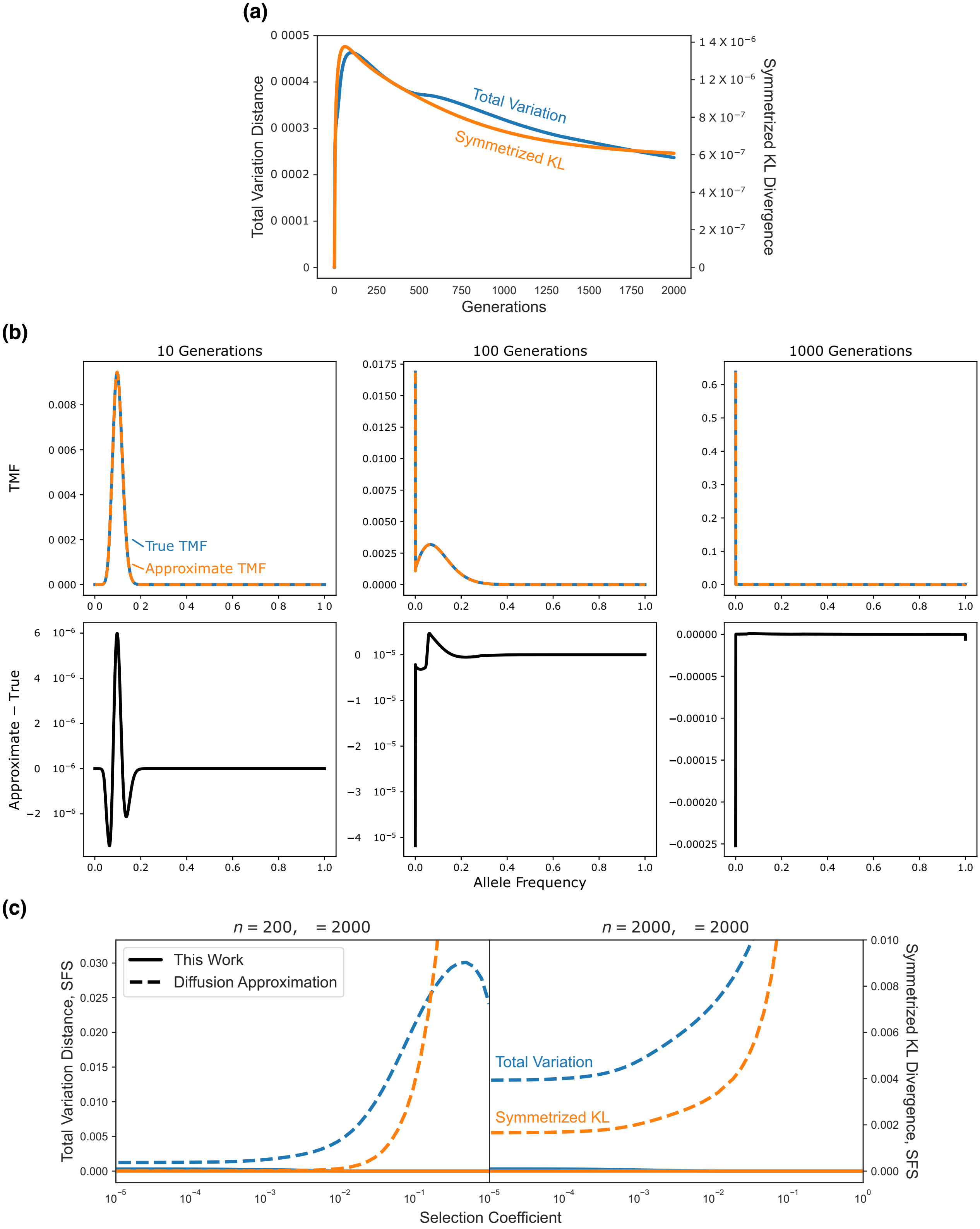

Fig. 4. Accuracy over multiple matrix–vector multiplications. a) Accuracy of the approximate algorithm for computing TMFs for a starting allele frequency of 0.1 over multiple generations in terms of total variation distance (left axis label) or symmetrized KL divergence (right axis label). b) Example TMFs at 10, 100, and 1,000 generations for an allele at an initial frequency of 0.1. The first row shows the exact TMFs under the DTWF model, and the approximation derived in this work. The second row shows the difference between the approximate TMF and the exact TMF. These differences are generally about 3 orders of magnitude smaller than the values of the TMF for the relevant frequencies. c) Accuracy of the approximate algorithm for computing normalized SFSs, as well as the commonly used diffusion approximation.

the results presented for a range of selection coefficients and sample sizes, assuming a constant population size of 2, 000 in Fig. 4c. We restricted ourselves to a population size of 2, 000 haploids so that we could exactly compute the ground truth by finding the equilibrium of the full DTWF transition matrix by solving a matrix equation. To compute the diffusion approximation at equilibrium we used the results presented in Bustamante et al. (2001) and used as a baseline in Krukov and Gravel (2021).

The diffusion approximation is expected to be good when the selection coefficient is $\lesssim 1/N$ and the sample size is $\lesssim\sqrt{N}$ (but see Melfi and Viswanath 2018b). In this regime, we see that both our approximation and the diffusion approximation accurately reconstruct the true normalized SFS, but with our approximation being about 4× more accurate in terms of total variation distance, and about 30× more accurate in terms of symmetrized KL. Yet, the diffusion approximation breaks down dramatically for large sample sizes or strong selection, while our approximation remains faithful. Indeed, for a selection coefficient of 0.01, when $n = 200$, our approximation is 250× more accurate than the diffusion approximation in terms of total variation, and 4,000× more accurate in terms of symmetrized KL. Similarly, when $n = N = 2,000$, even when the selection coefficient is zero, our approximation is about 45× more accurate in terms of total variation, and 3,800× more accurate in terms of symmetrized KL. Taken together, we see that our approximation is highly accurate across the full spectrum of sample sizes and selection coefficients.

Impact of mutation, selection, and demography on the DTWF model

There has recently been growing interest in using the frequency of loss-of-function variants (LoFs) in large-scale exome sequencing projects to estimate measures of gene constraint (Lek et al. 2016; Cassa et al. 2017; Weghorn et al. 2019; Karczewski et al. 2020; LaPolice and Huang 2023; Agarwal et al. 2023). LoFs are variants such as early stop codons, splice-disrupting variants, or frameshifts, that result in the gene failing to make a viable protein. To a first approximation, all LoFs within a gene have roughly the same strength of selection acting against them, as they all have similar effects on the production of functional protein. As such, LoFs are attractive for studying selection as we can pool information across all LoFs within a gene to estimate a single LoF selection coefficient for that gene.

Previous approaches have relied on deterministic approximations (Cassa et al. 2017), simulations (Weghorn et al. 2019; Agarwal et al. 2023), or ad hoc methods and models (Lek et al. 2016; Karczewski et al. 2020; LaPolice and Huang 2023) to infer selection coefficients from LoF data. These approaches have yielded widely used measures of gene constraint and important insights into the landscape of constraint on human genes. Yet, without more principled computational machinery for computing likelihoods, it can be difficult to estimate the gains in power we might expect to see in different datasets. For example, how does increasing the sample size affect our power to estimate selection coefficients? How does demography affect power? Does sampling from a population that has experienced recent growth affect power? Are some types of variants more informative than others? In this section, we use our machinery to answer these questions.

To understand how sample size, mutation rate, and demography interact to affect power for estimating selection coefficients, we considered a variety of each of these parameters. In particular, we considered sample sizes ranging from $n = 10$ diploids to $n = 300{,}000$ diploids, encompassing the range from small pilot studies in nonmodel organisms to biobank-scale datasets. To understand the impact of mutation rates and recurrent mutations, we considered a low mutation rate typical of transversions in humans (2.44×10^{-9} per generation) as well as a high mutation rate typical of methylated CpG sites in humans (1.25×10^{-7} per generation) (Karczewski et al. 2020). For demographies, we considered slight modifications of three demographies estimated from data from the 1000 Genomes Project (Consortium 2012)—one demography estimated using the Multiple Sequentially Markovian Coalescent (MSMC) (Schiffels and Durbin 2014) from individuals labeled by the 1000 Genomes Project as "Utah residents with Northern and Western European ancestry" (CEU) and two estimated from individuals labeled as "Yoruba in Ibadan, Nigera," (YRI) one inferred using MSMC (Schiffels and Durbin 2014), which we will refer to as simply the "YRI demography," and one inferred using Relate (Speidel et al. 2019), which we will refer to as "YRI (Speidel)." The CEU demography consists of a strong bottleneck corresponding to the out-of-Africa event, and recent explosive growth, whereas the YRI demography lacks a bottleneck and has remained roughly constant in size over time. The YRI (Speidel) demography lacks the deep bottleneck of the CEU demography and has more explosive recent growth (Fig. D1). See Appendix D for more details.

We used these different sets of mutation rates, sample sizes, and demographies in a DTWF model. Specifically, following previous work we focused on a diploid model of additive selection on LoFs (Cassa et al. 2017; Weghorn et al. 2019; Agarwal et al. 2023), where having one copy of the LoF variant results in a fitness reduction of s_{het}, while having two copies results in a fitness reduction of $s_{hom} := 2s_{het}$ (but with fitness lower bounded by 0 in the event that $s_{het} > 0.5$). Our computational machinery was developed for haploid populations and only tracks allele frequencies and not genotype frequencies. To approximate the diploid model of selection, we set the expected frequency in the next generation, $p(f)$, as

$$p(f) := \frac{(1 - s_{het})\tilde{f}(1 - \tilde{f}) + (1 - s_{hom})\tilde{f}^2}{(1 - \tilde{f})^2 + 2(1 - s_{het})\tilde{f}(1 - \tilde{f}) + (1 - s_{hom})\tilde{f}^2},$$

with

$$\tilde{f} = f + \mu(1 - f),$$

where f is the frequency of the LoF in the current generation, so that $\tilde{f}$ is the frequency following mutation at rate μ. Under strong selection, frequencies will generally be low, so we ignore back-mutation. That is, we assume that the mutation rate from the LoF allele to the non-LoF allele is zero. This model matches the expected frequency change under the diploid selection model assuming Hardy–Weinberg equilibrium (Gillespie 2004).

Without back-mutation, if the population ever fixes for the LoF allele, it will forever be stuck there. Yet, from any other frequency of the LoF allele (including 0), it is always possible through mutation or genetic drift for the LoF allele to fix (assuming $s_{het} < 1$). This implies that the equilibrium of this process is the degenerate state where the population is fixed for the LoF mutation, which is obviously not biologically realistic. One could instead turn to the commonly used infinite sites model, but this comes with two issues.

First, any particular site must be nonsegregating with probability one as the infinite sites model assumes an infinitesimally small per-site mutation rate balanced by an infinitely large mutational target size. This assumption may be realistic when considering mutations genome-wide, but certainly breaks down when looking at single LoFs, or even across LoFs within a single gene.

Second, since the mutation rate per site is infinitesimally small under the infinite sites approximation, the probability of recurrent mutation is also 0. Recurrent mutation in this context refers to the same allele being generated at the same site more than once via independent mutation events. For small sample sizes and small mutation rates, the probability of independent mutations happening at the same site is extremely small, explaining the popularity of the infinite sites model. Yet, for the CpG mutation rate we know that recurrent mutations are common and play an important role in shaping diversity (Harpak *et al.* 2016).

Instead of relying on the infinite sites approximation, our computational machinery allows us to easily condition the evolutionary dynamics of an allele at a single site on nonfixation. Essentially, when a new LoF allele enters the population, we ignore any scenarios where it drifts to fixation in the population. Looking backward in time, in a finite population alleles must at some point in the past either have been fixed or totally absent from the population. Since we are explicitly not allowing the LoF allele to have been fixed at any point in the past, there must have been some point in the past at which the LoF arose as a new mutation in a population monomorphic for the non-LoF allele. In this way, there is a well-defined notion of an ancestral allele and a derived allele. The DTWF model conditioned on nonfixation is well behaved and has a nontrivial equilibrium. Additionally, it allows us to easily model recurrent mutations and obtain a nonzero probability of an individual site being segregating. See Appendix C for more details surrounding this subtlety.

Before investigating the impact of selection, we wanted to see if modeling recurrent mutations was necessary for biobank-scale datasets under our models. Recently, analytical results for recurrent mutations in the coalescent (i.e. in the diffusion limit, and assuming neutrality) have been developed (Wakeley *et al.* 2023). Here, we also focus on the neutral case for simplicity, and our results are qualitatively similar to those in Wakeley *et al.* (2023). The approaches are complimentary: the results in Wakeley *et al.* (2023) are analytic, while ours must be obtained numerically. For our machinery it is no more difficult to consider cases with various types of selection, whereas obtaining coalescent-based results in the presence of selection would be difficult.

We considered something analogous to the SFS under our model—the probability of observing a given frequency of a derived allele conditioned on the site being segregating. We show the results for the CEU demography in Fig. 5, where we see that for large sample sizes, recurrent mutations have a large effect on the frequency spectrum, with singletons being almost half as likely under the CpG mutation rate compared to the transversion mutation rate. This is somewhat counterintuitive—one might expect that under a high mutation rate there would be more rare variation, and that *is* true in absolute terms as there are more segregating sites, but given that a site is segregating, rare variants actually become less likely under higher mutation rates. Indeed, at a sample size of $n = 300{,}000$ diploids, the probability that a CpG is segregating is 0.678, while for transversions it is only 0.022. At smaller sample sizes the impact of recurrent mutation is negligible for realistic mutation rates, with the probability that a site is segregating being 0.033 for CpGs and 0.0007 for transversions for a sample size of $n = 100$ diploids. The results for the MSMC-inferred YRI demography are qualitatively consistent (Fig. D2), although some of our modeling choices result in an unusual and interesting nonconvex frequency spectrum, which we discuss in detail in Appendix D.

We next turned to understanding the impact of mutation rates on using LoF frequencies to estimate s_{het}. To this end, we computed the likelihood of observing each possible frequency (including 0) in a given sample for a range of values of s_{het} ranging from well below the nearly neutral limit ($s_{het} = 10^{-6}$) all the way up to nearly lethal ($s_{het} \approx 1$). We show the results for the two mutation rates we considered for a sample of size 300,000 diploids from the CEU demography in Fig. 6. The results show that rare variants are weakly indicative of strong selection, but otherwise observing an LoF of a given frequency acts as a soft threshold on s_{het}. For example, a doubleton confidently rules out $s_{het} > 0.1$, but is otherwise essentially equally consistent with any value of s_{het}. Similarly, an LoF at 1% frequency rules out any $s_{het} > 0.002$, but is otherwise relatively uninformative. The results are qualitatively similar across the two mutation rates, but nonsegregating CpGs provide much more evidence in favor of strong selection than nonsegregating transversions, consistent with recent work by Agarwal and Przeworski (2021) showing empirically and via simulation that a nonsegregating CpG at similar sample sizes is enough to confidently reject neutrality.

To more precisely quantify how informative different sample sizes, datasets, or mutation rates are for estimating selection, we used the Fisher Information, $\mathcal{I}$. Fisher Information quantifies the expected curvature of the likelihood function at a given value of s_{het} and can be thought of as an effective sample size multiplier in terms of number of variants. In the DTWF model, information is additive across independent sites, so a setting with twice the Fisher Information would require half as many independent variants to achieve the same level of accuracy, roughly speaking. More formally, the Cramer–Rao lower bound from statistics shows that any unbiased estimator of s_{het} must have variance greater than $1/\mathcal{I}$. As such, the Fisher Information can be thought of as being inversely related to the variance of the best unbiased estimator of s_{het}. In our setting, the Fisher Information is defined as

$$\mathcal{I}(s) := \sum_{k=0}^{2n} \mathbb{P}\{k \text{ LoF alleles in sample} \mid s_{het} = s\} \times \left(\frac{\mathrm{d}}{\mathrm{d}t} \log \mathbb{P}\{k \text{ LoF alleles in sample} \mid s_{het} = t\} \Big|_{t=s} \right)^2,$$

which we can compute using the likelihood curves shown in Fig. 6 via numerical differentiation. Note that the Fisher Information depends on the parameterization of s_{het}, and here we compute the Fisher Information for $\log_{10} s_{het}$ to match the parameterization shown in Fig. 6. As a result, the Fisher Information can be thought of as being related to how many orders of magnitude the uncertainty in s_{het} should span.

We began by investigating the information content of CpGs and transversions for estimating s_{het}. For all of the demographies and all sample sizes we find that the information content of a CpG is always between 49× and 51.5× greater than that of a transversion. This makes intuitive sense as the mutation rate is about 51.2× higher for CpGs, indicating that we would expect CpGs to be segregating roughly 50× as often as transversions. CpGs are usually slightly less than 51.2× as informative as transversions across different values of s_{het}, indicating that there are some diminishing returns.

Next we turned to how information grows with sample size. Unlike standard statistical settings, sampling additional individuals does not provide completely independent information, and one might expect information to plateau as the sample size grows. Indeed, since individuals share a common genealogy, as additional individuals are added to a sample they are increasingly likely to be closely related to someone already in the sample, and

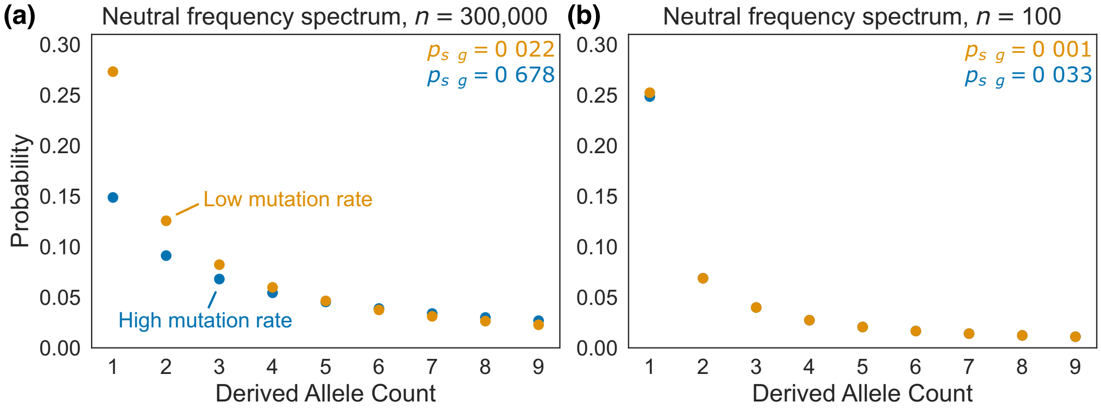

Fig. 5. Frequency spectra under the CEU demography for sample sizes $n = 300{,}000$ diploids a) and $n = 100$ diploids b) for a low mutation rate (2.44×10^{-9} per generation) and a high mutation rate (1.25×10^{-7} per generation). The low mutation roughly corresponds to the rate of a transversion in humans, while the high mutation rate rough corresponds to the rate of mutation at a methylated CpG. The probability of a site being segregating is reported as p_{seg}.

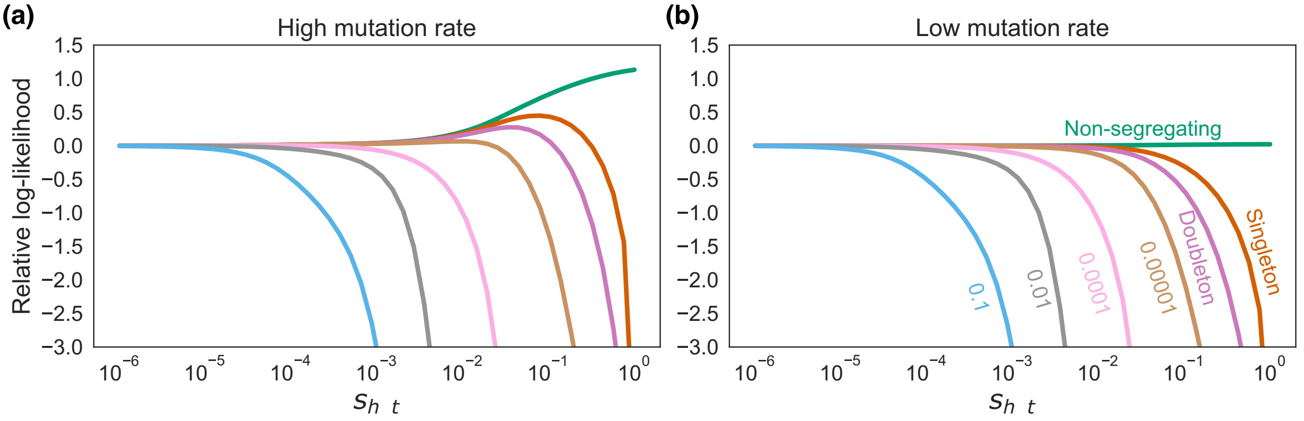

Fig. 6. Likelihoods for a sample of size $n = 300{,}000$ diploids from the CEU demography assuming either a high mutation rate of 1.25×10^{-7} per generation a) or a low mutation rate of 2.44×10^{-9} per generation b).

hence provide little additional information. This is borne out in our results (Fig. 7a), where we see that increasing sample sizes provide diminishing returns in terms of information. Yet, this effect is not uniform across the space of s_{het} values. Our results suggest that increasing sample sizes beyond the approximately 140,000 individuals in gnomAD (Karczewski *et al.* 2020) only provides additional information for the most extremely selected variants ($s_{het} > 0.02$). This highlights that there is a fundamental limit on how much we can hope to learn about the selective pressure on genes from LoF data alone—at current sample sizes we have already saturated the amount of information we might hope to obtain for a wide range of selection coefficients. Further increases in sample sizes will only help resolve the selection coefficients for the genes with the most extreme effects on fitness.

Finally, existing exome sequencing cohorts consist primarily of individuals that are genetically similar to the CEU individuals in the 1000 Genomes Project (Karczewski *et al.* 2020; Backman *et al.* 2021), raising questions of whether we might be able to better estimate selection by looking at samples of individuals who have experienced different demographic histories. For example, is it more informative to have a sample of a given size from a population that underwent the CEU demography or a population that underwent one of the YRI demographies? Interestingly, we find that the answer depends strongly on s_{het} and the sample size. For the smallest sample sizes (e.g. $n < 100$ diploids) samples from any of the demographies are comparably informative for selection coefficients above 0.001, but samples from either YRI demography are almost twice as informative for selection coefficients below 0.001 (Fig. 7b and c). The dominance of a sample from the YRI demographies for low selection coefficients remains across sample sizes, but as sample sizes increase, samples from the CEU demography become increasingly more informative for large selection coefficients relative to the MSMC-inferred YRI demography. For example, at a sample size of $n = 1{,}000$ diploids, a sample from the CEU demography is nearly twice as informative for an s_{het} of 0.01, and for a sample of size $n = 300{,}000$ a sample from the CEU demography is about 15 times as informative for an s_{het} of 0.1. In contrast, the YRI (Speidel) demography results in more information than the CEU demography across almost all values of s_{het} for sufficiently large sample sizes.

The relative Fisher Informations for samples can be understood in terms of the properties of the demographies. Variants under weak selection are older, and due to the out-of-Africa bottleneck, those variants will have experienced stronger drift under the CEU demography than the YRI demographies. Hence, samples from the CEU demography contain more "demographic noise" due to drift for such variants. Conversely, the recent explosion in population size in the CEU and YRI (Speidel) demographies results

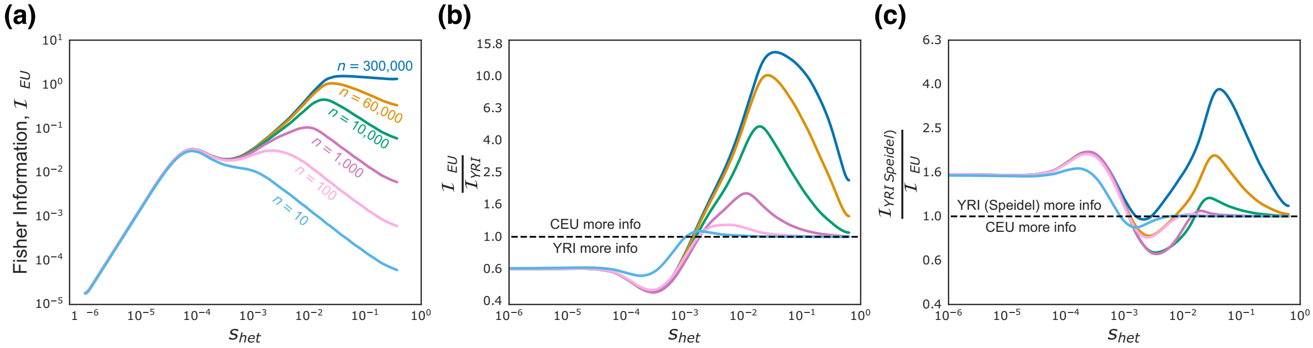

Fig. 7. a) Fisher Information as a function of s_{het} in samples from the CEU demography of different sample sizes. b) Fisher Information for a sample from the CEU demography relative to the Fisher Information for a sample of the same size from the YRI demography. Points above the dashed line indicate settings where the CEU sample provides more information, points from below the dashed line indicate settings where the YRI sample provides more information. c) Fisher Information for a sample from the YRI (Speidel) demography relative to the Fisher Information for a sample of the same size from the CEU demography.

in the opposite phenomenon for strongly selected variants which likely arose more recently, resulting in more information for large values of s_{het}. Finally, for small sample sizes there is little power to get at the rare, recent variants indicative of strong selection, regardless of demography, explaining why differences in information for large values of s_{het} only manifest at large sizes.

Discussion

Here, we presented an approach to approximate the transition matrix of the DTWF process that is provably accurate and allows us to compute likelihoods in $O(N)$ time. We showed that our approach can scale to population sizes in the tens of millions, and is highly accurate. Our approach relied on two key observations: the transition matrix of the DTWF process is approximately sparse, with only $O(\sqrt{N})$ entries contributing appreciably to the mass in each row, and the matrix is approximately low rank, where the matrix can be replaced by one with only $O(\sqrt{N})$ unique rows while incurring a small error.

We used our approach to understand how increasing sample sizes will help estimate the strength of selection acting against gene loss-of-function. We found that increasing sample sizes beyond those currently available will only provide additional information for the most strongly selected genes. For genes with anything weaker than the most extreme selection, current samples provide essentially as much information as can ever be obtained from LoF data from individuals closely related to the 1000 Genomes CEU sample.

Our approach may seem similar to choosing a discretization scheme for the PDE that describes the WF diffusion, but the approaches are distinct. The PDE discretization approach starts with the DTWF model, passes to a continuum limit to obtain a PDE, and then discretizes *that* continuous process. Yet, the WF diffusion is only valid for fixed frequencies as $N \to \infty$, indicating that in practice the continuous process is not a good approximation for frequencies close to 0 or 1. As such, any discretization of the WF diffusion must also be inaccurate near the boundaries. Instead, here we propose directly coarse graining the underlying discrete process without passing to a continuum limit.

In some ways, our approach is reminiscent of the scaling approach used in simulations (Adrion *et al.* 2020). In forward-in-time simulations, it can be computationally onerous to simulate a large population. To avoid this, one chooses a scaling factor, such as 10, and simulates a population 10× smaller, but increases the mutation rates, recombination rates, and strengths of selection by a factor of 10. Additionally, each generation in this scaled model counts for 10 generations in the unscaled model. This scaling is chosen so that the rescaled population converges to the same WF diffusion as the original process. Yet, this rescaling is only trustworthy for frequencies $\gg 1/N$. Here, we do not rescale parameters, but we do group states into "meta-states," and we group states more aggressively when the frequency is close to 0.5, and less aggressively for frequencies near 0 or 1. Whether a similar idea of frequency-adaptive rescaling could be incorporated into simulation to improve speed while remaining accurate is an interesting area for future research.

Another view of our method is that we are replacing a difficult set of transition distributions with a simpler set. This idea is very general and different approaches could be taken. For example, it may be possible to match the first several moments using only a very small number of nonzero entries. Such extremely sparse transition matrices could result in highly accurate and very computationally efficient approximations. The approach presented here is just one possibility in this vein, and exploring alternatives could be a fruitful direction for future research.

Our results are quite general, and can be readily extended to multiple alleles, multiple populations, or multiple loci. All of these can be treated as processes defined by a transition matrix of sub-Gaussian probability mass functions, and similar arguments to those used here can be applied to show that such transition matrices have approximately sparse rows, and are approximately low rank. These arguments should result in comparable speedups, but unfortunately, this direct approach of computing likelihoods using the forward transition matrix necessarily comes with a steep computational cost in these settings. For example, simply to list all of the possible configurations of a population of N individuals at two biallelic loci requires $O(N^3)$ time (Kamm *et al.* 2016). To list all of the possible configurations for 3 loci requires $O(N^7)$ time, and in general k loci requires $O(N^{(2^k-1)})$ time. There may be additional approximations that can be made in these cases, but simply approximating the transition matrix as we do here will not be enough to handle these more combinatorially difficult cases.

Throughout, we have assumed that the goal is to approximate the underlying DTWF model while maintaining computational efficiency. In general, however, no population will exactly follow any simple DTWF model—in many populations there will be fine-scale geographic population structure (Diaz-Papkovich *et al.*

2019), assortative mating (Yengo *et al.* 2018), overlapping generations, and so on. While these complications may make the DTWF model seem overly simplistic, the WF diffusion must be an even worse approximation as it also implies unrealistic family size distributions for large sample sizes (Melfi and Viswanath 2018b). In any case, our approach may also be useful for more complex models [e.g. general Cannings' exchangeable models (Cannings 1974; Ewens 2004)], as long as transitions have the two properties of being restricted (with high probability) to a small subset of the state space, and transition probability mass functions for nearby states being similar enough to be nearly indistinguishable. The extent to which these two properties are true will determine the extent of the speedup offered by our approach, and will depend on details of the underlying model. For example, the forward-in-time models that result in coalescents with multiple mergers (Pitman 1999; Eldon and Wakeley 2006) or simultaneous multiple mergers (Mohle and Sagitov 2001; Spence *et al.* 2016) often correspond to "sweepstakes reproduction" where a single individual may spawn a sizable fraction of the next generation. Under these models, a large sweepstakes reproduction event could cause an allele to dramatically change frequency in a single generation indicating the the transition density for any state is *not* approximately sparse, and the approach used in this paper would not result in a large speedup.

Here, we focused on the problem of computing the likelihood of observing a given number of derived alleles at present, but our speedups apply to time series data as well, which is frequently encountered in ancient DNA. Several methods have been developed that treat the true allele frequency at a given time as a hidden state in a hidden Markov model (HMM). This frequency then evolves through time according to the transition matrix of either the DTWF (Jewett *et al.* 2016), WF diffusion (Steinrücken *et al.* 2014), or some other approximation (Mathieson and Terhorst 2022), with sampled genotypes as the observations in the HMM. These HMMs have been particularly useful in estimating the strength of natural selection acting on individual loci, and our results can be used in these methods to speed up computations while directly approximating the DTWF model.

Our implementation is in `pytorch` (Paszke *et al.* 2019), which allows for backward mode automatic differentiation, enabling the computation of gradients of functions of the likelihood with respect to selection coefficients or mutation rates. Unfortunately, backward mode automatic differentiation requires storing the entire computation graph in memory. In our setting, this corresponds to storing representations of the approximate transition matrices at each generation, which may become memory intensive in models where the nonequilibrium portion spans many generations. Indeed, throughout this paper we resorted to using numerical approximations to the gradient to avoid these issues. Since our likelihood computation essentially just involves repeated matrix–vector multiplication, one may view it as a very deep neural network with linear activations, and backward mode automatic differentiation proves to be memory intensive in those applications as well (Gao *et al.* 2020). Our approach is also mathematically similar to using discretization to integrate a linear ODE forward in time, another application which essentially boils down to repeated matrix–vector multiplication. In that setting powerful methods have been developed which essentially solve the ODE forward to calculate the likelihood and then backward to obtain gradients, which avoids the need to store the computation graph in memory (Chen *et al.* 2018). Extending this approach to our setting is a promising approach to obtain gradients without resorting to numerical approximation. Yet, one of the most interesting parameters of the model, the population size, is necessarily discrete in the DTWF model, and hence is not differentiable. Approximations such as the Straight-Through Estimator (Bengio *et al.* 2013) could get around this, but their accuracy would require careful investigation.

While our approach makes it feasible to accurately compute likelihoods under the DTWF model, the runtime can still be quite onerous. For example, computing the likelihood of observing all possible allele frequency for a single selection coefficient took ~10 min for the demographies considered here. For application such as the inference of selection coefficients where there are a small number of parameters, practitioners can precompute likelihoods along a grid of values in parallel. Yet, for applications such as demographic inference with many parameters, this grid approach is infeasible and one would need to repeatedly compute likelihoods over the course of optimization. In such scenarios, a runtime of ~10 min per likelihood could be prohibitive. Bayesian optimization (Snoek *et al.* 2012) is tailored to optimizing functions that are expensive to evaluate and versions that take advantage of parallelism (Snoek *et al.* 2012, Section 3.3) are promising candidates for using our approach for demographic inference.

As modern datasets approach sample sizes of hundreds of thousands to millions, new scalable approaches are needed in population genetics. This onslaught of data is a blessing, but more work like this—developing provably accurate, scalable approaches—is needed to keep up and allow us to extract useful insights from these ever growing sample sizes. Yet, care should be taken as our results show that larger sample sizes are not always helpful. For the problem of estimating selection coefficients, larger sample sizes will never provide less information, but for many genes they will not provide more information.

Data availability

Code for computing likelihoods is available at https://github.com/jeffspence/fastDTWF. The authors affirm that all data necessary for confirming the conclusions of the article are present within the article and figures.

Acknowledgments

We would like to thank Ian Holmes for discussions about coarse graining Markov chains and Paul Jenkins for discussions about conditioning on nonfixation. We would like to thank Clemens Weiß and Amy Ko for helpful discussions and valuable feedback on the manuscript.

Funding

This work was supported by NIH grant R01HG011432.

Conflicts of interest

The author(s) declare no conflict of interest.

Literature cited

Adell JA, Jodrá P. 2006. Exact Kolmogorov and total variation distances between some familiar discrete distributions. J Inequal Appl. 2006:1–8. doi:10.1155/JIA/2006/64307

Adrion JR, Cole CB, Dukler N, Galloway JG, Gladstein AL, Gower G, Kyriazis CC, Ragsdale AP, Tsambos G, Baumdicker F, *et al* 2020.

A community-maintained standard library of population genetic models. eLife. 9:e54967. doi:10.7554/eLife.54967

Agarwal I, Fuller ZL, Myers SR, Przeworski M. 2023. Relating pathogenic loss-of function mutations in humans to their evolutionary fitness costs. eLife. 12:e83172. doi:10.7554/eLife.83172

Agarwal I, Przeworski M. 2021. Mutation saturation for fitness effects at human CpG sites. eLife. 10:e71513. doi:10.7554/eLife.71513

Backman JD, Li AH, Marcketta A, Sun D, Mbatchou J, Kessler MD, Benner C, Liu D, Locke AE, Balasubramanian S, *et al* 2021. Exome sequencing and analysis of 454,787 UK Biobank participants. Nature. 599:628–634. doi:10.1038/s41586-021-04103-z

Bengio Y, Léonard N, Courville A. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. arXiv, arXiv:1308.3432, preprint: not peer reviewed.

Bhaskar A, Clark AG, Song YS. 2014. Distortion of genealogical properties when the sample is very large. Proc Natl Acad Sci USA. 111: 2385–2390. doi:10.1073/pnas.1322709111

Bhaskar A, Wang YR, Song YS. 2015. Efficient inference of population size histories and locus-specific mutation rates from large-sample genomic variation data. Genome Res. 25:268–279. doi:10.1101/gr.178756.114

Bustamante CD, Wakeley J, Sawyer S, Hartl DL. 2001. Directional selection and the site-frequency spectrum. Genetics. 159: 1779–1788. doi:10.1093/genetics/159.4.1779

Cannings C. 1974. The latent roots of certain Markov chains arising in genetics: a new approach, I. Haploid models. Adv Appl Probab. 6: 260–290. doi:10.2307/1426293

Cassa CA, Weghorn D, Balick DJ, Jordan DM, Nusinow D, Samocha KE, O'Donnell-Luria A, MacArthur DG, Daly MJ, Beier DR, *et al.* 2017. Estimating the selective effects of heterozygous protein-truncating variants from human exome data. Nat Genet. 49: 806–810. doi:10.1038/ng.3831

Chen RT, Rubanova Y, Bettencourt J, Duvenaud DK. 2018. Neural ordinary differential equations. Adv Neural Inf Process Syst. 31: 6571–6583.

Consortium GP. 2012. An integrated map of genetic variation from 1,092 human genomes. Nature. 491:56. doi:10.1038/nature11632

Computing (STOC '87). New York (NY): Association for Computing Machinery. p. 1–6. doi:10.1145/28395.28396

Diaz-Papkovich A, Anderson-Trocmé L, Ben-Eghan C, Gravel S. 2019. UMAP reveals cryptic population structure and phenotype heterogeneity in large genomic cohorts. PLoS Genet. 15:e1008432. doi:10.1371/journal.pgen.1008432

Eldon B, Wakeley J. 2006. Coalescent processes when the distribution of offspring number among individuals is highly skewed. Genetics. 172:2621–2633. doi:10.1534/genetics.105.052175

Evans SN, Shvets Y, Slatkin M. 2007. Non-equilibrium theory of the allele frequency spectrum. Theor Popul Biol. 71:109–119. doi:10.1016/j.tpb.2006.06.005

Ewens WJ. 2004. Mathematical Population Genetics: Theoretical Introduction. Vol. 27. New York (NY): Springer.

Fu YX. 2006. Exact coalescent for the Wright–Fisher model. Theor Popul Biol. 69:385–394. doi:10.1016/j.tpb.2005.11.005

Gao Y, Liu Y, Zhang H, Li Z, Zhu Y, Lin H, Yang M. 2020. Estimating GPU memory consumption of deep learning models. Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020). New York (NY): Association for Computing Machinery. p. 1342–1352. doi:10.1145/3368089.3417050

Gao Z, Moorjani P, Sasani TA, Pedersen BS, Quinlan AR, Jorde LB, Amster G, Przeworski M. 2019. Overlooked roles of dna damage and maternal age in generating human germline mutations. Proc Natl Acad Sci USA. 116:9491–9500. doi:10.1073/pnas.1901259116

Gao Z, Wyman MJ, Sella G, Przeworski M. 2016. Interpreting the dependence of mutation rates on age and time. PLoS Biol. 14: e1002355. doi:10.1371/journal.pbio.1002355

Gibbs AL, Su FE. 2002. On choosing and bounding probability metrics. Int Stat Rev. 70:419–435. doi:10.1111/j.1751-5823.2002.tb00178.x

Gillespie JH. 2004. Population Genetics: A Concise Guide. Baltimore (MD): JHU Press.

Gutenkunst RN, Hernandez RD, Williamson SH, Bustamante CD. 2009. Inferring the joint demographic history of multiple populations from multidimensional SNP frequency data. PLoS Genet. 5: e1000695. doi:10.1371/journal.pgen.1000695

Harpak A, Bhaskar A, Pritchard JK. 2016. Mutation rate variation is a primary determinant of the distribution of allele frequencies in humans. PLoS Genet. 12:e1006489. doi:10.1371/journal.pgen.1006489

Hoeffding W. 1963. Probability inequalities for sums of bounded random variables. J Am Stat Assoc. 58:13–30. doi:10.1080/01621459.1963.10500830

Jansen S, Kurt N. 2014. On the notion(s) of duality for Markov processes. Probab Surv. 11:59–120. doi:10.1214/12-PS206

Jewett EM, Steinrücken M, Song YS. 2016. The effects of population size histories on estimates of selection coefficients from time-series genetic data. Mol Biol Evol. 33:3002–3027. doi:10.1093/molbev/msw173

Jónsson H, Sulem P, Kehr B, Kristmundsdottir S, Zink F, Hjartarson E, Hardarson MT, Hjorleifsson KE, Eggertsson HP, Gudjonsson SA, *et al.* 2017. Parental influence on human germline de novo mutations in 1,548 trios from Iceland. Nature. 549:519–522. doi:10.1038/nature24018

Jouganous J, Long W, Ragsdale AP, Gravel S. 2017. Inferring the joint demographic history of multiple populations: beyond the diffusion approximation. Genetics. 206:1549–1567. doi:10.1534/genetics.117.200493

Kamm JA, Spence JP, Chan J, Song YS. 2016. Two-locus likelihoods under variable population size and fine-scale recombination rate estimation. Genetics. 203:1381–1399. doi:10.1534/genetics.115.184820

Kamm J, Terhorst J, Durbin R, Song YS. 2020. Efficiently inferring the demographic history of many populations with allele count data. J Am Stat Assoc. 115:1472–1487. doi:10.1080/01621459.2019.1635482

Kamm JA, Terhorst J, Song YS. 2017. Efficient computation of the joint sample frequency spectra for multiple populations. J Comput Graph Stat. 26:182–194. doi:10.1080/10618600.2016.1159212

Karczewski KJ, Francioli LC, Tiao G, Cummings BB, Alföldi J, Wang Q, Collins RL, Laricchia KM, Ganna A, Birnbaum DP, *et al.* 2020. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature. 581:434–443. doi:10.1038/s41586-020-2308-7

Kim BY, Huber CD, Lohmueller KE. 2017. Inference of the distribution of selection coefficients for new nonsynonymous mutations using large samples. Genetics. 206:345–361. doi:10.1534/genetics.116.197145

Kingman JFC. 1982. The coalescent. Stoch Process their Appl. 13: 235–248. doi:10.1016/0304-4149(82)90011-4

Koch E, Novembre J. 2017. A temporal perspective on the interplay of demography and selection on deleterious variation in humans. G3. 7:1027–1037. doi:10.1534/g3.117.039651

Krone SM, Neuhauser C. 1997. Ancestral processes with selection. Theor Popul Biol. 51:210–237. doi:10.1006/tpbi.1997.1299

Krukov I, de Sanctis B, de Koning APJ. 2016. Wright–Fisher exact solver (WFES): scalable analysis of population genetic models without simulation or diffusion theory. Bioinformatics. 33:1416–1417. doi:10.1093/bioinformatics/btw802

Krukov I, Gravel S. 2021. Taming strong selection with large sample sizes. bioRxiv.2021.03. doi:10.1101/2021.03.30.437711

LaPolice TM, Huang YF. 2023. An unsupervised deep learning framework for predicting human essential genes from population and functional genomic data. BMC Bioinform 24:347. doi:10.1186/s12859-023-05481-z

Lek M, Karczewski KJ, Minikel EV, Samocha KE, Banks E, Fennell T, O'Donnell-Luria AH, Ware JS, Hill AJ, Cummings BB, *et al.* 2016. Analysis of protein-coding genetic variation in 60,706 humans. Nature. 536:285–291. doi:10.1038/nature19057

Mathieson I, Terhorst J. 2022. Direct detection of natural selection in bronze age Britain. Genome Res. 32:2057–2067. doi:10.1101/gr.276862.122

Melfi A, Viswanath D. 2018a. Single and simultaneous binary mergers in Wright–Fisher genealogies. Theor Popul Biol. 121:60–71. doi:10.1016/j.tpb.2018.04.001

Melfi A, Viswanath D. 2018b. The Wright–Fisher site frequency spectrum as a perturbation of the coalescent's. Theor Popul Biol. 124: 81–92. doi:10.1016/j.tpb.2018.09.005

Mohle M, Sagitov S. 2001. A classification of coalescent processes for haploid exchangeable population models. Ann Probab. 29: 1547–1562. doi:10.1214/aop/1015345761

Nagaev SV. 1965. Some limit theorems for large deviations. Theory Probab Appl. 10:214–235. doi:10.1137/1110027

Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, *et al.* 2019. Pytorch: an imperative style, high-performance deep learning library. Adv Neural Inf Process Syst. 32:8026–8037.

Paul JS, Song YS. 2012. Blockwise HMM computation for large-scale population genomic inference. Bioinformatics. 28:2008–2015. doi:10.1093/bioinformatics/bts314

Pitman J. 1999. Coalescents with multiple collisions. Ann Probab. 27: 1870–1902. doi:10.1214/aop/1022874819

Polanski A, Kimmel M. 2003. New explicit expressions for relative frequencies of single-nucleotide polymorphisms with application to statistical inference on population growth. Genetics. 165: 427–436. doi:10.1093/genetics/165.1.427

Roos B. 2001. Binomial approximation to the Poisson binomial distribution: the Krawtchouk expansion. Theory Probab Appl. 45: 258–272. doi:10.1137/S0040585X9797821X

Sargsyan O, Wakeley J. 2008. A coalescent process with simultaneous multiple mergers for approximating the gene genealogies of many marine organisms. Theor Popul Biol. 74:104–114. doi:10.1016/j.tpb.2008.04.009

Sasani TA, Pedersen BS, Gao Z, Baird L, Przeworski M, Jorde LB, Quinlan AR. 2019. Large, three-generation human families reveal post-zygotic mosaicism and variability in germline mutation accumulation. eLife. 8:e46922. doi:10.7554/eLife.46922

Sawyer SA, Hartl DL. 1992. Population genetics of polymorphism and divergence. Genetics. 132:1161–1176. doi:10.1093/genetics/132.4.1161

Schiffels S, Durbin R. 2014. Inferring human population size and separation history from multiple genome sequences. Nat Genet. 46: 919–925. doi:10.1038/ng.3015

Snoek J, Larochelle H, Adams RP. 2012. Practical Bayesian optimization of machine learning algorithms. In: Pereira F, Burges C, Bottou L, Weinberger K, editors. Advances in Neural Information Processing Systems. Vol. 25. Red Hook (NY): Curran Associates, Inc. p. 2951–2959.

Song YS, Steinrücken M. 2012. A simple method for finding explicit analytic transition densities of diffusion processes with general diploid selection. Genetics. 190:1117–1129. doi:10.1534/genetics.111.136929

Speidel L, Forest M, Shi S, Myers SR. 2019. A method for genome-wide genealogy estimation for thousands of samples. Nat Genet. 51: 1321–1329. doi:10.1038/s41588-019-0484-x

Spence JP, Kamm JA, Song YS. 2016. The site frequency spectrum for general coalescents. Genetics. 202:1549–1561. doi:10.1534/genetics.115.184101

Spence JP, Song YS. 2019. Inference and analysis of population-specific fine-scale recombination maps across 26 diverse human populations. Sci Adv. 5:eaaw9206. doi:10.1126/sciadv.aaw9206

Steinrücken M, Bhaskar A, Song YS. 2014. A novel spectral method for inferring general diploid selection from time series genetic data. Ann Appl Stat. 8:2203.

Steinrücken M, Jewett EM, Song YS. 2016. Spectraltdf: transition densities of diffusion processes with time-varying selection parameters, mutation rates and effective population sizes. Bioinformatics. 32:795–797. doi:10.1093/bioinformatics/btv627

Steinrücken M, Kamm J, Spence JP, Song YS. 2019. Inference of complex population histories using whole-genome sequences from multiple populations. Proc Natl Acad Sci USA. 116:17115–17120. doi:10.1073/pnas.1905060116

Steinrücken M, Wang YR, Song YS. 2013. An explicit transition density expansion for a multi-allelic Wright–Fisher diffusion with general diploid selection. Theor Popul Biol. 83:1–14. doi:10.1016/j.tpb.2012.10.006

Strassen V. 1969. Gaussian elimination is not optimal. Numer Math. 13:354–356. doi:10.1007/BF02165411

Tataru P, Simonsen M, Bataillon T, Hobolth A. 2016. Statistical inference in the Wright–Fisher model using allele frequency data. Syst Biol. 66:e30–e46.

Terhorst J, Kamm JA, Song YS. 2017. Robust and scalable inference of population history from hundreds of unphased whole genomes. Nat Genet. 49:303–309. doi:10.1038/ng.3748

Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, *et al* 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods. 17:261–272. doi:10.1038/s41592-019-0686-2

Wakeley J, Fan WTL, Koch E, Sunyaev S. 2022. 2023. Recurrent mutation in the ancestry of a rare variant. Genetics 223:iyad049. doi:10.1093/genetics/iyad049

Wakeley J, Takahashi T. 2003. Gene genealogies when the sample size exceeds the effective size of the population. Mol Biol Evol. 20:208–213. doi:10.1093/molbev/msg024

Weghorn D, Balick DJ, Cassa C, Kosmicki JA, Daly MJ, Beier DR, Sunyaev SR. 2019. Applicability of the mutation–selection balance model to population genetics of heterozygous protein-truncating variants in humans. Mol Biol Evol. 36:1701–1710. doi:10.1093/molbev/msz092

Yengo L, Robinson MR, Keller MC, Kemper KE, Yang Y, Trzaskowski M, Gratten J, Turley P, Cesarini D, Benjamin DJ, *et al.* 2018. Imprint of assortative mating on the human genome. Nat Hum Behav. 2:948–954. doi:10.1038/s41562-018-0476-3

Zeng T, Spence JP, Mostafavi H, Pritchard JK. 2023. Bayesian estimation of gene constraint from an evolutionary model with gene features. bioRxiv. doi:10.1101/2023.05.19.541520

Živković D, Steinrücken M, Song YS, Stephan W. 2015. Transition densities and sample frequency spectra of diffusion processes with selection and variable population size. Genetics. 200: 601–617. doi:10.1534/genetics.115.175265

Appendix A: Formal theoretical results and proofs

Notation

Throughout, we will use the notation $B_{N,p}$ for a Binomial distribution with sample size N and success probability p. We will also restrict our attention to what we define as *ordered Binomial transition matrices*, of which many DTWF models are special cases. We say that a matrix, $\mathbf{M} \in \mathbb{R}^{(N_1+1)\times(N_2+1)}$ is a *Binomial transition matrix* if each row of $\mathbf{M}$ is the probability mass function of a Binomial distribution. That is, for each i the ith row of $\mathbf{M}$ is the probability mass function corresponding to a B_{N_2,p_i} distribution. Furthermore, we say that $\mathbf{M}$ is an *ordered* Binomial transition matrix if it is a Binomial transition matrix with the further property that the rows are ordered by success probability; that is, $p_0 \le p_1 \le \cdots \le p_{N_1}$.

Construction of approximate transition matrix

The main goal of this section is to prove that we can accurately approximate any Binomial transition matrix with a highly structured matrix for which matrix–vector multiplication is much faster than standard. The overall crux of this proof is 2-fold.

First, nearby rows in a Binomial transition matrix are close in total variation distance, indicating that we can replace one row with a copy of a nearby row while only incurring a small error. To prove this, we will show that the total variation distance between $B_{N,p}$ and $B_{N,p'}$ is small so long as p and p' are close in a particular sense. This will allows us to partition the interval [0, 1] into $O(\sqrt{N})$ blocks such that for any p and p' in the same block the corresponding Binomial distributions are guaranteed to be close in total variation distance. This in turn shows that *any* Binomial transition matrix for a population of size N can be replaced by a matrix with only $O(\sqrt{N})$ unique rows while controlling the rowwise error.

Second, we will use a classical result to show that the tails of the Binomial distribution are incredibly light—it is very unlikely to sample a value far away from the mean of a Binomial distribution, and in particular, one incurs only a small approximation error by only considering the possibility of sampling something within $O(\sqrt{N})$ of the mean. This will allow us to replace each row of the Binomial transition matrix by a sparse vector with only $O(\sqrt{N})$ nonzero entries, while only incurring a small approximation error.

To start showing that we can partition [0, 1] into $O(\sqrt{N})$ blocks where Binomial distributions with success probabilities in the same block have bounded total variation distance, we begin with the block that includes 0.

Lemma A1. *For any* $p \le 1-(1-\varepsilon)^{1/N}$,

$$d_{TV}(B_{N,0}, B_{N,p}) \le \varepsilon.$$

In particular, for fixed ε, p *can be as large as* $O(\frac{1}{N})$ *while maintaining a total variation distance to a Binomial distribution with success probability 0 at most* ε.

Proof. Let $X \sim B_{N,0}$ and $Y \sim B_{N,p}$. Note that X must be 0 with probability 1, and cannot take any other value. The total variation is then, by definition,

$$\begin{aligned} d_{TV}(B_{N,0}, B_{N,p}) &= \frac{1}{2}\sum_{k=0}^{N}|\mathbb{P}\{X=k\} - \{Y=k\}| \\ &= \frac{1}{2}(1-\mathbb{P}\{Y=0\}) + \frac{1}{2}\sum_{k=1}^{N}\mathbb{P}\{Y=k\} \\ &= \frac{1}{2}(1-\mathbb{P}\{Y=0\}) + \frac{1}{2}(1-\mathbb{P}\{Y=0\}) \\ &= 1-\mathbb{P}\{Y=0\} \\ &= 1-(1-p)^{N}. \end{aligned}$$

This is obviously monotonically increasing in p, and solving for p we obtain that

$$p = 1-(1-d_{TV}(B_{N,0}, B_{N,p}))^{1/N},$$

so to obtain a total variation distance at most ε we need

$$p \le 1-(1-\varepsilon)^{1/N}.$$

Finally, rewriting $(1-\varepsilon)^{1/N}$ as $\exp\left(\frac{1}{N}\log(1-\varepsilon)\right)$ and using the convergent series expansion, we see

$$1-(1-\varepsilon)^{1/N} = \frac{-\log(1-\varepsilon)}{N} + O\left(\frac{1}{N^2}\right).$$

□

The following lemma, stated in Adell and Jodrá (2006) and proved in Roos (2001) bounds the total variation distance between Binomial distributions with success probabilities away from 0 and 1.

Lemma A2. *(Adell and Jodrá 2006, Equations (2.15) and (2.16)).* For any $p \in (0, 1)$ and any $\delta \in (0, 1-p)$,

$$d_{TV}(B_{N,p}, B_{N,p+\delta}) \le \frac{\sqrt{e}}{2}\frac{\tau(\delta)}{(1-\tau(\delta))^2},$$

where

$$\tau(\delta) := \delta\sqrt{\frac{N+2}{2p(1-p)}}.$$

For our purposes, we just need to know how far away p' can be from p before incurring an unacceptable total variation distance, which we can obtain by loosening the above bound and rearranging.

Lemma A3. *For any* $p \in (0, 1/2)$, *there exists a constant* c_ε, *that depends on* ε *but not on* p *or* N *such that for any* $\delta \in [0, c_\varepsilon\sqrt{\frac{p}{N}})$ *we have*

$$d_{TV}(B_{N,p}, B_{N,p+\delta}) \le \varepsilon.$$

Proof. First, note that for any $p \in (0, 1/2)$ and $N \geq 1$ we have

$$\sqrt{\frac{N+2}{2p(1-p)}} \leq \sqrt{\frac{3N}{p}}.$$

Letting $x := \delta\sqrt{\frac{3N}{p}}$, we have by Lemma A2 that

$$d_{\mathrm{TV}}(B_{N,p}, B_{N,p+\delta}) \leq \frac{\sqrt{e}}{2}\frac{x}{(1-x)^2}.$$

The right-hand side is obviously monotonically increasing in x on $[0, 1)$ from a value of 0 at $x = 0$ to infinity as x approaches 1. Furthermore, the equation does not contain p or N (except in the definition of x). Therefore, there exists a c'_ε independent of p and N such that when $x = c'_\varepsilon$ the right-hand side is ε, and hence for any $x \leq c'_\varepsilon$ we have that $d_{\mathrm{TV}}(B_{N,p}, B_{N,p+\delta}) \leq \varepsilon$. Using our definition of x and solving for δ completes the proof. □

With these Lemmas in place, we can now prove our result on partitioning $[0, 1]$ such that Binomial distributions with success probabilities in the same block have bounded total variation distance.

Lemma A4. *For fixed ε, there exist $O(\sqrt{N})$ breakpoints $0 = p_0 < p_1 < p_2 < \cdots < p_K = 1$ such that for any p and p' within in the same interval (i.e. there exists an i such that $p, p' \in [p_i, p_{i+1}]$) we have*

$$d_{\mathrm{TV}}(B_{N,p}, B_{N,p'}) \leq \varepsilon.$$

Proof. Note that by symmetry, we only need to consider partitioning the space $[0, 1/2]$ using $O(\sqrt{N})$ breakpoints. By Lemma A3, we can control the total variation distance of all distributions between a breakpoint p_k and p_{k+1} by taking

$$p_{k+1} = c_\varepsilon\sqrt{\frac{p_k}{N}} + p_k. \tag{A1}$$

Therefore, we have that the total variation distance is less than ε between any Binomials of size N with success probabilities between p_k and p_{k+1}.

Now, we will prove by induction that there exists a constant, α_ε, that depends on ε but not on N, such that

$$p_k \geq \frac{\alpha_\varepsilon k^2}{N}. \tag{A2}$$

The base case of p_1 is handled by Lemma A1. Now, suppose that Equation (A2) holds for p_k, then, by Equation (A1) and the inductive hypothesis,

$$\begin{aligned} p_{k+1} &\geq \frac{k c_\varepsilon\sqrt{\alpha_\varepsilon}}{N} + \frac{\alpha_\varepsilon k^2}{N} \\ &= \frac{\alpha_\varepsilon(k+1)^2}{N} + \left(\frac{c_\varepsilon\sqrt{\alpha_\varepsilon}}{N} - \frac{2\alpha_\varepsilon}{N}\right)k - \frac{\alpha_\varepsilon}{N} \\ &\geq \frac{\alpha_\varepsilon(k+1)^2}{N}, \end{aligned}$$

where the last line follows by noting that $k \geq 1$ and taking α_ε to be at least as small as $c_\varepsilon^2/9$.

This proves Equation (A2). Then, to partition $[0, 1/2]$ we can compute breakpoints using the recursion in Equation (A1) until we reach the first breakpoint larger than 1/2. By Equation (A2), we need at most

$$\left\lceil \sqrt{\frac{N}{2\alpha_\varepsilon}} \right\rceil$$

breakpoints, which is $O(\sqrt{N})$, to partition the space, completing the proof. □

We now turn to the task of showing that for a given Binomial distribution almost all of the mass is on outcomes within $O(\sqrt{N})$ of the mean. This result follows straightforwardly from Hoeffding's celebrated inequality (Hoeffding 1963), which we include for completeness

Lemma A5. *(Hoeffding's Inequality Hoeffding (1963, Theorem 1)). Let $X \sim B_{N,p}$, then for $k \leq Np$,*

$$\mathbb{P}\{X \leq k\} \leq \exp\left(-2N\left(p - \frac{k}{N}\right)^2\right)$$

and for $k \geq Np$,

$$\mathbb{P}\{X \geq k\} \leq \exp\left(-2N\left(p - \frac{k}{N}\right)^2\right).$$

Lemma A6. Let

$$k_{min}(N, p, \varepsilon) = \left\lfloor Np - \sqrt{N}\sqrt{\frac{-\log\frac{\varepsilon}{2}}{2}} \right\rfloor$$

$$k_{max}(N, p, \varepsilon) = \left\lceil Np + \sqrt{N}\sqrt{\frac{-\log\frac{\varepsilon}{2}}{2}} \right\rceil$$

and define $\widetilde{B}^\varepsilon_{N,p}$ as the distribution obtained by conditioning $B_{N,p}$ to take values in

$$[\max\{k_{min}(N, p, \varepsilon) + 1, 0\}, \min\{k_{max}(N, p, \varepsilon) - 1, N\}].$$

Then,

$$d_{\mathrm{TV}}(\widetilde{B}^\varepsilon_{N,p}, B_{N,p}) \leq \varepsilon$$

and the mass function for $\widetilde{B}^\varepsilon_{N,p}$ contains only $O(\sqrt{N})$ nonzero entries.

Proof. That the mass function of $\widetilde{B}^\varepsilon_{N,p}$ contains only $O(\sqrt{N})$ nonzero entries is obvious from its construction. To bound the total variation distance, let $X \sim B_{N,p}$, and let $Y \sim \widetilde{B}^{\varepsilon}_{N,p}$. By construction,

$$\begin{aligned} d_{TV}(\widetilde{B}^{\varepsilon}_{N,p}, B_{N,p}) &= \frac{1}{2}\mathbb{P}\{X \le k_{\min}\} + \frac{1}{2}\mathbb{P}\{X \ge k_{\max}\} \\ &\quad + \frac{1}{2}\sum_{k=k_{\min}+1}^{k_{\max}-1} \mathbb{P}\{Y = k\} - \mathbb{P}\{X = k\} \\ &= \frac{1}{2}\mathbb{P}\{X \le k_{\min}\} + \frac{1}{2}\mathbb{P}\{X \ge k_{\max}\} \\ &\quad + \frac{\frac{1}{\mathbb{P}\{k_{\min} < X < k_{\max}\}} - 1}{2}\sum_{k=k_{\min}+1}^{k_{\max}-1} \mathbb{P}\{X = k\} \\ &= \frac{1}{2}\mathbb{P}\{X \le k_{\min}\} + \frac{1}{2}\mathbb{P}\{X \ge k_{\max}\} \\ &\quad + \frac{1 - \mathbb{P}\{k_{\min} < X < k_{\max}\}}{2} \\ &= \mathbb{P}\{X \le k_{\min}\} + \mathbb{P}\{X \ge k_{\max}\}, \end{aligned}$$

where we dropped the dependence of $k_{\min}$ and $k_{\max}$ on N, p, and ε for notational convenience. Then, by Lemma A5,

$$\begin{aligned} \mathbb{P}\{X \le k_{\min}\} &\le \exp\left(-2N\left(p - \frac{k_{\min}}{N}\right)^2\right) \\ &\le \frac{\varepsilon}{2}, \end{aligned}$$

where the second line follows from our choice of $k_{\min}$. An analogous computation shows that

$$\mathbb{P}\{X \ge k_{\max}\} \le \frac{\varepsilon}{2},$$

completing the proof. □

The following lemma will be used to show that as long as a Binomial transition matrix is ordered then we can assign the success probabilities for each of its rows to a set of blocks in linear time.

Lemma A7. *Consider $0 \le v_1 \le v_2 \le \cdots \le v_N \le 1$, and a partition of the space $[0, 1]$ defined by breakpoints $0 = p_0 < p_1 < \cdots < p_{K-1} < p_K = 1$. We may compute index sets $\mathcal{S}_1, \ldots, \mathcal{S}_K$ in $O(N + K)$ time such that for all k for each $i \in \mathcal{S}_k$ we have that $p_{k-1} \le v_i \le p_k$, with $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_N = \{1, \ldots, N\}$ and $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ for all $i \neq j$.*

Proof. We begin with v_1 and we find the first k such that p_k that is at least as large as v_1. We may then search from v_2 onward until we find the first j such that v_j is larger than p_k. If no such j exists, then set j to be $N + 1$. We assign $1, \ldots, j - 1$ to index set $\mathcal{S}_k$. This is a valid assignment as we have $p_{k-1} \le v_1 \le \ldots \le v_{j-1} \le p_k$. We may then repeat this process—we next find the first k_2 such that p_{k_2} is at least as large as v_j, and we find the first j_2 such that v_{j_2} is larger than p_{k_2} (or if such a j_2 does not exist, set j_2 to $N + 1$), assigning $j, \ldots, j_2 - 1$ to $\mathcal{S}_{k_2}$. We can repeat this process until all of the vs have been assigned to an index set. Since both the vs and ps are sorted we can do these searches starting where the previous search left off, and once we reach the end of both the vs and the ps we have assigned all of the vs to an index set. Thus, we only have to consider each v and p $O(1)$ times, resulting in a total runtime of $O(N + K)$. □

Combining the previous lemmas, we can construct a highly structured matrix that accurately approximates an ordered Binomial transition matrix in $O(N)$ time.

Proposition A1. *Let $\mathbf{M} \in \mathbb{R}^{(N_1+1)\times(N_2+1)}$ be an ordered Binomial transition matrix. We can build a representation of a matrix $\widetilde{\mathbf{M}}$ with the following properties:*

- *$\widetilde{\mathbf{M}}$ has $O(\sqrt{N_2})$ unique rows,*
- *Each row of $\widetilde{\mathbf{M}}$ has at most $O(\sqrt{N_2})$ nonzero elements,*
- *$\|\mathbf{M}^{\mathsf{T}} - \widetilde{\mathbf{M}}^{\mathsf{T}}\|_1 \le \varepsilon$.*

Furthermore, we can construct this representation in $O(N_1 + N_2)$ time and store it in $O(N_1 + N_2)$ space.

Proof. Since $\mathbf{M}$ is an ordered Binomial transition matrix, each row is a probability mass function of the form B_{N_2,p_i} with nondecreasing p_i. By Lemma A4, we can partition the space $[0, 1]$ into $O(\sqrt{N_2})$ blocks such that for any p_i, p_j in the same bucket the total variation distance between the rows is $\varepsilon/4$. We can assign each of the Binomial probability mass functions to these blocks in $O(N_1 + \sqrt{N_2})$ time by Lemma A7. Each row of the matrix with an index in the same index set can then be replaced by a Binomial probability mass function with an arbitrary "representative" success probability contained in that block. Since each row is getting replaced by a row corresponding to a distribution with total variation distance less than $\varepsilon/4$, we have that the ℓ_1 distance between each row is less than $\varepsilon/2$. Finally, by Lemma A6 we can replace the distribution for each representative row, by an $O(\sqrt{N_2})$ sparse version while only incurring a further total variation distance of $\varepsilon/4$, resulting in a further ℓ_1 distance of $\varepsilon/2$. By the triangle inequality, each final row is then at most ε away from the original row in ℓ_1 distance.

Since there are only $O(\sqrt{N_2})$ unique rows and each each of these have only $O(\sqrt{N_2})$ elements, once we have assigned rows to their representative rows constructing this sparse representation only requires $O(N_2)$ time, resulting in an overall runtime of $O(N_1 + N_2)$. Representing $\widetilde{\mathbf{M}}$ requires $O(N_1 + N_2)$ space, as storing the $O(\sqrt{N_2})$ unique rows, each with $O(\sqrt{N_2})$ nonzero entries requires $O(N_2)$ space, and then we must also store an index for each row in $\widetilde{\mathbf{M}}$ indicating which representative row to use, requiring $O(N_1)$ space. □

The requirement in Proposition A1 that the Binomial transition matrix be ordered is so that we can determine which rows can be replaced with which representative rows in linear time. In the absence of such information (i.e. if the Binomial success probabilities for each row of the matrix are arbitrary) then we need $O(N \log N)$ time to determine the representative rows to use for each row.

The following lemma shows that matrix–vector products can be made substantially faster for matrices with a limited number of unique rows where each of those rows are sparse.

Lemma A8. *Let $\mathbf{M} \in \mathbb{R}^{N\times P}$ be a matrix with n unique rows, where each row has at most s nonzero elements. Furthermore, suppose we have index sets $\mathcal{S}_1, \ldots, \mathcal{S}_n$, with $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n = \{1, \ldots, N\}$ and $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ for all $i \neq j$ with the property that if rows k and ℓ are in the same index set than those rows of $\mathbf{M}$ are identical. Then, for any vector $\mathbf{v} \in \mathbf{R}^N$ the matrix–vector product $\mathbf{M}^{\mathsf{T}}\mathbf{v}$ can be computed in $O(N + P + ns)$ time.*

Proof. Let $\mathbf{M}_i$ be the ith row of $\mathbf{M}$. For the desired matrix product we need to compute

$$\mathbf{M}^{\mathsf{T}}\mathbf{v} = \sum_{i=1}^{N} (\mathbf{M}_i)^{\mathsf{T}}\mathbf{v}_i$$

Since the index sets cover all of the indices with no overlaps, we can rewrite the above sum as:

$$\mathbf{M}^{\top}\mathbf{v} = \sum_{k=1}^{n}\sum_{i\in\mathcal{S}_k}(\mathbf{M}_i)^{\top}\mathbf{v}_i.$$

Let $i_1, \ldots, i_n$ be arbitrarily chosen elements from each of the n index sets. Then, by noting that if i is in $\mathcal{S}_k$, we have by assumption $(\mathbf{M}_i)^{\top} = (\mathbf{M}_{i_k})^{\top}$, which allows us to pull the inner summation inward:

$$\mathbf{M}^{\top}\mathbf{v} = \sum_{k=1}^{n}(\mathbf{M}_{i_k})^{\top}\sum_{i\in\mathcal{S}_k}\mathbf{v}_i. \tag{A3}$$

The inner sum is now a sum of scalars, so for a particular k it costs $O(|\mathcal{S}_k|)$ to compute. To compute it across all k thus costs $O(N)$ time. Then, since each $\mathbf{M}_{i_k}$ only contains s nonzeros, we can multiply each by a scalar, and then sum up all n of them in $O(ns + P)$ time. To see this, note that we can initialize a "running total" vector of P zeros, and then for each $\mathbf{M}_{i_k}^{\top}$ we only need to update the running total in the s entries at which $\mathbf{M}_{i_k}$ is nonzero. □

Combining Proposition A1 and Lemma A8, we immediately arrive at our main result.

Theorem A1. *Let $\mathbf{M} \in \mathbb{R}^{(N_1+1)\times(N_2+1)}$ be an ordered Binomial transition matrix. We can replace $\mathbf{M}$ by a matrix $\widetilde{\mathbf{M}}$ such that $\|\mathbf{M}^{\top} - \widetilde{\mathbf{M}}^{\top}\|_1 \le \varepsilon$, and such that computing matrix vector products of the form $\widetilde{\mathbf{M}}^{\top}\mathbf{v}$ requires $O(N_1 + N_2)$ time.*

Proof. Each row of $\mathbf{M}$ is a probability mass function of the form B_{N_2,p_i} with nondecreasing p_i. By Proposition A1, we can approximate this matrix by a matrix with at most $O(\sqrt{N_2})$ unique rows, each with at most $O(\sqrt{N_2})$ nonzero entries, while maintaining $\|\mathbf{M}^{\top} - \widetilde{\mathbf{M}}^{\top}\|_1 \le \varepsilon$, and we can construct a representation of this matrix in $O(N_1 + N_2)$ time. By Lemma A8, we can perform matrix–vector multiplication with such a matrix in $O(N_1 + N_2)$ time. □

We note that these results are all in terms of total variation distance. We prove results about how our approximation affects the first two moments of the transition distributions in Appendix E.

Sample likelihoods

In this section, we discuss how to efficiently obtain the likelihood of observing k A alleles in a sample of size n from the vector of probabilities of observing different numbers of A alleles in a population of size N. While this may seem like a substantially different problem, we show that similar considerations to speeding up matrix–vector multiplication for Binomial transition matrices can be applied to this subsampling problem as well. In particular, it is generally assumed that the sample of size n is drawn without replacement from the population of size N. Suppose that there are K A alleles in the population, then the probability of drawing some number of A alleles in a sample of size n is determined by the Hypergeometric distribution with parameters N, K, and n, which we will denote in this section by $H_{N,K,n}$. Since we do not observe the frequency of the A allele in the population, we should integrate over this latent variable. Thus, if $\mathbf{v}_{\text{sample}} \in \mathbb{R}^{n+1}$ is the vector of sample probabilities, and $\mathbf{v} \in \mathbb{R}^{N+1}$ is the vector of population probabilities, then,

$$\mathbf{v}_{\text{sample}} = \mathbf{S}^{\top}\mathbf{v},$$

where $\mathbf{S}$ is a "sampling matrix" where the kth row is the probability mass function corresponding to the distribution $H_{N,k,n}$.

Surprisingly, our results about Binomial transition matrices also apply in modified forms to Hypergeometric sampling matrices. In particular, we show below that such matrices have rows that are approximately $O(\sqrt{n})$ sparse, and furthermore that such matrices are also close to matrices with $O(\sqrt{n})$ unique rows. These two tricks will allow us to compute sampling probabilities from population probabilities in $O(N)$ time.

To show that the rows of $\mathbf{S}$ are approximately sparse, we again use Hoeffding's celebrated inequality. In his original paper, Hoeffding shows that his bounds on the tails of Binomial distributions also hold for Hypergeometric distributions (but phrased as sampling with and without replacement) (Hoeffding 1963). We include the result below for completeness.

Lemma A9. *(Adapted from Hoeffding (1963, Theorem 4)). Let $X \sim H_{N,K,n}$, then for $k \le nK/N$,*

$$\mathbb{P}\{X \le k\} \le \exp\left(-2n\left(\frac{K}{N} - \frac{k}{n}\right)^2\right)$$

and for $k \ge nK/N$,

$$\mathbb{P}\{X \ge k\} \le \exp\left(-2n\left(\frac{K}{N} - \frac{k}{n}\right)^2\right).$$

This lemma immediately implies an analogous sparsity result to Lemma A6. As the proof is essentially identical to the proof of that lemma we omit it.

Lemma A10. *Let*

$$k_{\min}(N, K, n, \varepsilon) = \left\lfloor \frac{nK}{N} - \sqrt{n}\sqrt{\frac{-\log\frac{\varepsilon}{2}}{2}} \right\rfloor$$

$$k_{\max}(N, K, n, \varepsilon) = \left\lceil \frac{nK}{N} + \sqrt{n}\sqrt{\frac{-\log\frac{\varepsilon}{2}}{2}} \right\rceil$$

and define $\widetilde{H}^{\varepsilon}_{N,K,n}$ as the distribution obtained by conditioning $H_{N,K,n}$ to take values between $k_{\min} + 1$ and $k_{\max} - 1$. Then,

$$d_{\mathrm{TV}}(\widetilde{H}^{\varepsilon}_{N,K,n}, H_{N,K,n}) \le \varepsilon,$$

and the mass function for $\widetilde{H}^{\varepsilon}_{N,K,n}$ contains only $O(\sqrt{n})$ nonzero entries.

We will also need a result analogous to Lemma A2, showing that nearby Hypergeometric distributions are close in total variation distance. It turns out that the Hypergeometric case is slightly more delicate than the Binomial case. Our results rely on the assumption that we are not too close to sampling the entire population in that we will assume that $n \le \alpha N$ for some $\alpha < 1$, and we will consider the $N \to \infty$ limit. Note that this still encompasses cases where we sample say 99% of the population, but rules out some pathological asymptotics such as having $n = N - N^{1-\varepsilon}$ where the proportion of the population sampled increases with the population size.

Before proving our general result about the total variation distance between different Hypergeometric distributions, we first prove a "one-step" inequality.

Lemma A11. *Suppose that $n \leq \alpha N$ for some $\alpha < 1$. Furthermore, suppose that $K \leq N/2$. Then, for any N larger than some finite N_0,*

$$d_{\mathrm{TV}}(\mathrm{H}_{N,K,n}, \mathrm{H}_{N,K+1,n}) \leq c_\alpha \sqrt{\frac{n}{KN}},$$

where c_α is a universal constant that depends on α but is independent of n, N, and K.

Proof. We start with the definition of total variation distance. Letting p_k be the probability that a random variable distributed as $\mathrm{H}_{N,K,n}$ is k, and letting q_k be the analogous quantity for $\mathrm{H}_{N,K+1,n}$ we have

$$d_{\mathrm{TV}}(\mathrm{H}_{N,K,n}, \mathrm{H}_{N,K+1,n}) = \frac{1}{2}\sum_{k=0}^{n} |p_k - q_k|.$$

It is difficult to directly bound this sum sufficiently tightly, so instead we would like to convert it into an expectation by noting that

$$|p_k - q_k| = q_k\left|\frac{p_k}{q_k} - 1\right|$$

but an issue arises in that in general $\mathrm{H}_{N,K,n}$ and $\mathrm{H}_{N,K+1,n}$ can each put zero mass on an event where the other does not (i.e. neither is absolutely continuous with respect to the other), making the above potentially ill-defined. In particular, q_k is zero but p_k is still positive when $k = n + K - N$.

Fortunately, we can use Lemma A9 to show that these events do not contribute substantially to the total variation distance. That is, if we take

$$k_{\min} := \max\left\{n\frac{K}{N} - \sqrt{\frac{n\log(KN/n)}{2}}, 0\right\},$$

we can show that $k_{\min} > n + K - N$ by noting that

$$\begin{aligned}
k_{\min} - (n + K - N) &\geq N - K - n + n\frac{K}{N} - \sqrt{\frac{n\log(KN/n)}{2}}\\
&= n\left(\frac{K}{N} - 1\right) + N - K - \sqrt{\frac{n\log(KN/n)}{2}}\\
&\geq \alpha(K - N) + N - K - \sqrt{N\log N}\\
&= (1-\alpha)(N - K) - \sqrt{N\log N}\\
&\geq \frac{(1-\alpha)N}{2} - \sqrt{N\log N},
\end{aligned}$$

where we used that $n \leq \alpha N$ and $K \leq \frac{N}{2}$ by assumption. The $\sqrt{N\log N}$ term is lower order than $(1-\alpha)N/2$, so $k_{\min} - (n + K - N) \geq c'_\alpha N > 0$ for some $c'_\alpha > 0$ so long as N is large enough. This allows us to avoid the issue of dividing by zero, while only incurring an $o(\sqrt{n/KN})$ error term on the total variation, by Lemma A9:

$$d_{\mathrm{TV}}(\mathrm{H}_{N,K,n}, \mathrm{H}_{N,K+1,n}) \leq o\left(\sqrt{\frac{n}{KN}}\right) + \frac{1}{2}\sum_{k=k_{\min}}^{n} q_k\left|\frac{p_k}{q_k} - 1\right|,$$

where we abuse notation and define the summand to be zero when p_k and q_k are both zero. Plugging in the values of p_k and q_k shows that when q_k is nonzero,

$$\begin{aligned}
\left|\frac{p_k}{q_k} - 1\right| &= \left|\frac{k(N+1) - n(K+1)}{(K+1)(N - K - n + k)}\right|\\
&\leq \frac{1}{(K+1)(N - K - n + k_{\min})}|k(N+1) - n(K+1)|.
\end{aligned}$$

This allows us to bound the total variation distance in terms of an expectation of a random variable X distributed as $\mathrm{H}_{N,K+1,n}$.

$$\begin{aligned}
&d_{\mathrm{TV}}(\mathrm{H}_{N,K,n}, \mathrm{H}_{N,K+1,n})\\
&\quad\leq o\left(\sqrt{\frac{n}{KN}}\right) + \frac{\sum_{k=k_{\min}}^{n} q_k|k(N+1) - n(K+1)|}{2(K+1)(N - K - n + k_{\min})}\\
&\quad\leq o\left(\sqrt{\frac{n}{KN}}\right) + \frac{\sum_{k=0}^{n} q_k|k(N+1) - n(K+1)|}{2(K+1)(N - K - n + k_{\min})}\\
&\quad= o\left(\sqrt{\frac{n}{KN}}\right) + \frac{\mathbb{E}|X(N+1) - n(K+1)|}{2(K+1)(N - K - n + k_{\min})}\\
&\quad\leq o\left(\sqrt{\frac{n}{KN}}\right) + \frac{\sqrt{\mathbb{E}[(X(N+1) - n(K+1))^2]}}{2(K+1)(N - K - n + k_{\min})}
\end{aligned}$$

where the final line follows from Jensen's inequality. This expectation only relies on the first two moments of a $\mathrm{H}_{N,K+1,n}$ random variable, and so may be readily, although tediously computed:

$$\begin{aligned}
&\mathbb{E}\left[\left(X(N+1) - n(K+1)\right)^2\right]\\
&\quad= \frac{n(K+1)}{N(N-1)}\left\{K(n(N+3) - (N+1)^2)\right.\\
&\qquad\left.+ (N-1)((N+1)^2 - n(N+2))\right\}\\
&\quad\leq cnKN
\end{aligned}$$

for some universal constant c, where we naively bounded all appearances of n and K in the curly braces by factors of N. Noting from above that for N sufficiently large $N - K - n + k_{\min} \geq c'_\alpha N$ for some constant c'_α independent of N, K, and n, results in the desired bound:

$$\begin{aligned}
d_{\mathrm{TV}}(\mathrm{H}_{N,K,n}, \mathrm{H}_{N,K+1,n}) &\leq o\left(\sqrt{\frac{n}{KN}}\right)\\
&\quad+ \frac{\sqrt{cnKN}}{2(K+1)(N - K - n + k_{\min})}\\
&\leq c_\alpha\sqrt{\frac{n}{KN}}.
\end{aligned}$$

□

It then becomes quite easy to combine this one-step lemma to obtain something analogous to Lemma A3 but for Hypergeometric distributions.

Lemma A12. *Suppose that $n \leq \alpha N$ for some $\alpha < 1$ and that $K \leq \frac{N}{2}$. There exists a constant $c_{\alpha,\varepsilon}$ that depends on α and ε but not N, K, or n such that for any nonnegative integer $\ell \leq \min\{c_{\alpha,\varepsilon}\sqrt{KN/n}, \frac{N}{2} - K\}$*

we have

$$d_{\mathrm{TV}}(H_{N,K,n}, H_{N,K+\ell,n}) \leq \varepsilon.$$

Proof. Total variation distance is a metric, and hence by the triangle inequality,

$$\begin{aligned} d_{\mathrm{TV}}(H_{N,K,n}, H_{N,K+\ell,n}) &\leq \sum_{j=K}^{K+\ell-1} d_{\mathrm{TV}}(H_{N,j,n}, H_{N,j+1,n}) \\ &\leq \sum_{j=K}^{K+\ell-1} c_\alpha \sqrt{\frac{n}{jN}} \\ &\leq \ell c_\alpha \sqrt{\frac{n}{KN}} \\ &\leq c_{\alpha,\varepsilon} c_\alpha, \end{aligned}$$

where the second line followed from Lemma A11. Choosing $c_{\alpha,\varepsilon} \leq \varepsilon / c_\alpha$ completes the proof. □

We also need to consider where the first breakpoint can be.

Lemma A13. *Suppose that* $n \leq \alpha N$. *Then, for any* K *such that* $K/N \leq (1-\alpha)[1-(1-\varepsilon)^{1/n}]$,

$$d_{\mathrm{TV}}(H_{N,0,n}, H_{N,K,n}) \leq \varepsilon.$$

In particular, for fixed ε, K/N *may be as large as* $O(1/n)$ *while maintaining a total variation distance of at most* ε *to the* $H_{N,0,n}$ *distribution.*

Proof. If we let $X \sim H_{N,K,n}$, then, as in the proof of Lemma A1, the total variation distance between the distributions can be written as

$$d_{\mathrm{TV}}(H_{N,0,n}, H_{N,K,n}) = 1 - \mathbb{P}\{X = 0\}.$$

A quick calculation shows that

$$\begin{aligned} \mathbb{P}\{X = 0\} &= \frac{(N-K)(N-K-1)\cdots(N-K-n+1)}{N(N-1)\cdots(N-n+1)} \\ &\geq \left(1 - \frac{K}{N-n+1}\right)^n. \end{aligned}$$

Using that $n \leq \alpha N$, we obtain

$$\mathbb{P}\{X = 0\} \geq \left(1 - \frac{K}{(1-\alpha)N}\right)^n.$$

Therefore,

$$d_{\mathrm{TV}}(H_{N,0,n}, H_{N,K,n}) \leq 1 - \left(1 - \frac{K}{(1-\alpha)N}\right)^n.$$

Solving for K/N, one obtains that the total variation distance is bounded by ε so long as

$$K/N \leq (1-\alpha)[1-(1-\varepsilon)^{1/n}].$$

The term on the right-hand side is the same, up to the factor of $(1-\alpha)$ as in the Binomial case. Therefore, it is $O(1/n)$ following the asymptotic argument in Lemma A1. □

With Lemmas A12 and A13 in hand, we can prove a result analogous to Lemma A4 using similar techniques used in the proof of that result.

Lemma A14. *Suppose that* $n \leq \alpha N$ *for some* $\alpha < 1$. *For fixed* ε, *there exist* $O(\sqrt{n})$ *breakpoints* $0 = p_0 < p_1 < p_2 < \cdots < p_M = 1$ *such that for any* K *and* K' *with* K/N *and* K'/N *being in the same block (i.e. there exists an* i *such that* $K/N, K'/N \in [p_i, p_{i+1}]$) *we have*

$$d_{\mathrm{TV}}(H_{N,K,n}, H_{N,K',n}) \leq \varepsilon.$$

Proof. The proof follows immediately from the proof of Lemma A4, by noting that we may replace Lemma A3 by Lemma A12 and we may replace Lemma A1 by Lemma A13. □

Combining these lemmas, we obtain our main approximation result for the sampling matrix.

Proposition A2. *Let* $\mathbf{S} \in \mathbb{R}^{N+1,n+1}$ *be a Hypergeometric sampling matrix. We can build a representation of a matrix* $\widetilde{\mathbf{S}}$ *with the following properties:*

- $\widetilde{\mathbf{S}}$ *has* $O(\sqrt{n})$ *unique rows,*
- *Each row of* $\widetilde{\mathbf{S}}$ *has at most* $O(\sqrt{n})$ *nonzero elements,*
- $\|\mathbf{S}^\top - \widetilde{\mathbf{S}}^\top\|_1 \leq \varepsilon$.

Furthermore, we can construct this representation in $O(N)$ time and store it in $O(N)$ space.

Proof. The result follows immediately from an argument analogous to the proof of Proposition A1. □

Finally, we note that this approximate matrix satisfies the properties of Lemma A8 allowing us to compute sample likelihoods from population likelihoods in $O(N)$ time.

Appendix B: Proof of the representation of the 1-operator norm

For completeness we include a proof that the 1-operator norm of a matrix is the max of the ℓ_1 norm across columns. This is a standard, well-known result.

Proof. Consider an $N \times P$ matrix A. We will complete the proof in two steps—first we will show that $\|A\|_1$ is at least as large as the max of the ℓ_1 norm across columns, then we will show that $\|A\|_1$ is no larger than the max of the ℓ_1 norm across columns.

Without loss of generality, assume that the first column of A has the largest ℓ_1 norm. Consider the vector e_1, which has a 1 for its first entry and zero for all other entries. Clearly $\|e_1\|_1 = 1$, and so

$$\begin{aligned} \|A\|_1 &= \sup_{x : \|x\|_1} \|Ax\|_1 \\ &\geq \| A e_1 \|_1 \end{aligned}$$

but $\|Ae_1\|_1$ is just the ℓ_1 norm of the first column of A which is the column with the largest ℓ_1 norm.

To prove the other direction, we will need to consider the columns of A which we will write as $A_{\cdot,1}, \ldots, A_{\cdot,P}$. We see that for any x with $\|x\|_1 = 1$,

$$\begin{aligned}\|Ax\|_1 &= \left\|\sum_{j=1}^{P} A_{\cdot,j}x_j\right\|_1 \\ &\leq \sum_{j=1}^{P} \|A_{\cdot,j}x_j\|_1 \\ &= \sum_{j=1}^{P} |x_j|\|A_{\cdot,j}\|_1 \\ &\leq \max_k \|A_{\cdot,k}\|_1 \sum_{j=1}^{P} |x_j| \\ &= \max_k \|A_{\cdot,k}\|_1,\end{aligned}$$

where the first inequality followed from the triangle inequality. □

Appendix C: Practical considerations

In this section, we will consider some practical aspects of using Binomial transition matrices in the context of the DTWF model. First, in *Faster repeated matrix–vector products* section, we will discuss a practical implementation detail that allows for faster likelihood computations when the underlying DTWF dynamics do not change too frequently. Then, in *Computing the stationary distribution* section, we discuss how to efficiently compute equilibria under our model. We then discuss two aspects of the DTWF model that are useful in practice—an infinite sites version of the DTWF model to compute frequency spectra (*Infinite sites* section), and a version of the DTWF model conditioned on nonfixation, which is similar in spirit to the infinite sites model, but is conceptually cleaner when considering models with recurrent mutation (*Conditioning on nonfixation* section).

Faster repeated matrix–vector products

As discussed in the main text, we may need to repeatedly perform matrix vector products to compute likelihoods under the DTWF model for nonequilibrium populations. If the underlying dynamics of the DTWF model do not change too frequently, then we can obtain substantial computational savings by considering a "condensed" transition matrix and then repeatedly squaring that condensed matrix. In particular, we consider the case where we need to compute $(\mathbf{M}^\top)^k\mathbf{v}$ for large k. In principle, we could use Theorem A1 to approximate this by k matrix-vector products as $\widetilde{\mathbf{M}}^\top \cdots \widetilde{\mathbf{M}}^\top\mathbf{v}$ (evaluated right to left), which would cost $O(kN)$ time, but if k is large we can speed this up substantially. One can view our fast multiplication algorithm for $\widetilde{\mathbf{M}}^\top\mathbf{v}$, Equation (3), as consisting of two steps. First, we project $\mathbf{v}$ into a $O(\sqrt{N})$ dimensional space by summing all of the entries of $\mathbf{v}$ that correspond to each of the $O(\sqrt{N})$ unique rows of $\widetilde{\mathbf{M}}$, and then we multiply the resulting vector by the transpose of a matrix where we only keep nonredundant rows of $\widetilde{\mathbf{M}}$, which we will call $\mathring{\mathbf{M}}$. If we were to repeat this process, we would then project the vector $\widetilde{\mathbf{M}}^\top\mathbf{v}$ to the same $O(\sqrt{N})$ dimensional space. If we write this projection operation as a matrix, $\boldsymbol{\Pi}$, then

$$\widetilde{\mathbf{M}}^\top\widetilde{\mathbf{M}}^\top\mathbf{v} = \mathring{\mathbf{M}}^\top\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top\boldsymbol{\Pi}\mathbf{v}.$$

In general,

$$(\widetilde{\mathbf{M}}^\top)^k\mathbf{v} = \mathring{\mathbf{M}}^\top(\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top)^{k-1}\boldsymbol{\Pi}\mathbf{v}$$

and the trick is to note that

$$(\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top) \in \mathbb{R}^{O(\sqrt{N})\times O(\sqrt{N})}$$

is a square matrix. As a result, we can repeatedly square $\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top$, with each squaring taking $O(N^{3/2})$ time, allowing us to compute $\mathring{\mathbf{M}}^{k-1}$ in $O(N^{3/2}\log k)$ time. In principle, this could be reduced substantially using faster matrix–matrix multiplication algorithms such as Strassen's algorithm [reduces runtime to $\approx O(N^{1.4037}\log k)$] (Strassen 1969) or the Coppersmith–Winograd algorithm [reduces runtime to $\approx O(N^{1.188}\log k)$] (Coppersmith and Winograd 1987). Additionally, one could diagonalize the transition matrix and then compute matrix powers, which would require $O(N^{3/2} + \sqrt{N}\log k)$ time. For simplicity and numerical stability, we stick with the naive matrix–matrix multiplication algorithm, resulting in a runtime of $O(N^{3/2}\log k)$. If k is $O(N^{\epsilon+1/2})$ or larger then this provides a faster algorithm. Similar tricks have been used in population genetics in the context of coalescent hidden Markov models (Paul and Song 2012; Terhorst *et al.* 2017; Steinrücken *et al.* 2019).

Computing the stationary distribution

Since we compute likelihoods forward in time, we must assume that at some point in the past the distribution of allele frequencies is known, and then use the DTWF transition matrix to integrate that distribution forward in time to the present. A natural choice is to assume that the population was at equilibrium at some point in the past, and to then compute the corresponding stationary distribution. By definition, when the system is at equilibrium, it is unchanged by the dynamics of the process, resulting in the following matrix equation:

$$\mathbf{M}^\top\mathbf{v}_{\mathrm{eq}} = \mathbf{v}_{\mathrm{eq}}(\mathbf{M}^\top - \mathbf{I})\mathbf{v}_{\mathrm{eq}} = \mathbf{0}.$$

One could in principle solve this matrix equation (with the constraint that $\mathbf{v}_{\mathrm{eq}}$ sums to one), but the naive strategy to obtain a solution costs $O(N^3)$ time. We might hope that by simply replacing $\mathbf{M}$ by our approximation, $\widetilde{\mathbf{M}}$ we might be able to solve this equation faster. While there are solvers that can take advantage of the sparsity of $\widetilde{\mathbf{M}}$ (Virtanen *et al.* 2020), there are no solvers that can also take advantage of the fact $\widetilde{\mathbf{M}}$ only has a small number of unique rows. Here, we propose two solutions, both of which rely on the ideas presented in *Faster repeated matrix–vector products* section.

One solution is to solve for the equilibrium of the "condensed" dynamics. In the notation of *Faster repeated matrix–vector products* section, we solve for $\mathring{\mathbf{v}}_{\mathrm{eq}}$ in

$$\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top\mathring{\mathbf{v}}_{\mathrm{eq}} = \mathring{\mathbf{v}}_{\mathrm{eq}},$$

which requires $O(N^{3/2})$ time. We then claim that $\mathring{\mathbf{M}}^\top\mathring{\mathbf{v}}_{\mathrm{eq}}$ is an equilibrium of $\widetilde{\mathbf{M}}$. To see this, note that

$$\begin{aligned}\widetilde{\mathbf{M}}^\top\left(\mathring{\mathbf{M}}^\top\mathring{\mathbf{v}}_{\mathrm{eq}}\right) &= \mathring{\mathbf{M}}^\top\boldsymbol{\Pi}\mathring{\mathbf{M}}^\top\mathring{\mathbf{v}}_{\mathrm{eq}} \\ &= \mathring{\mathbf{M}}^\top\mathring{\mathbf{v}}_{\mathrm{eq}}\end{aligned}$$

showing that $\mathring{\mathbf{M}}^{\mathsf{T}}\mathring{\mathbf{v}}_{\text{eq}}$ is invariant under the dynamics of $\widetilde{\mathbf{M}}$.

This approach requires a particular choice of representative success probabilities when constructing $\mathring{\mathbf{M}}$. In principle, we would like to choose success probabilities using our moment-matching approach, as described in the main text, but that requires knowing the equilibrium frequencies—exactly the object we wish to calculate. In practice, we find that using an iterative method where we use some initial guess of the equilibrium frequencies to construct $\mathring{\mathbf{M}}$, solve the above equation to obtain a new guess for the equilibrium frequencies, reconstruct $\mathring{\mathbf{M}}$, and so on, works well, resulting in errors on the order of machine precision after two or three iterations.

An alternative approach is to use the power method, which essentially approximates the equilibrium of a Markov chain by running the dynamics for a long time. That is,

$$\mathbf{v}_{\text{eq}} \approx (\widetilde{\mathbf{M}}^{\mathsf{T}})^{k}\mathbf{v}$$

for any initial distribution, $\mathbf{v}$, and taking k to be large. This is exactly the setting of *Faster repeated matrix–vector products* section, and so we may use those results to speed up this computation. In practice, we can choose the initial $\mathbf{v}$ to be close to the true equilibrium by using analytical solutions to the Wright–Fisher diffusion.

Infinite sites

The mutation rate at any given position in the genome can be vanishingly small. For example, in humans the premodern effective population size was on the order of 10,000 (Schiffels and Durbin 2014), and per-base mutation rates are, on average, about 10^{-8} per generation (Jónsson *et al.* 2017). Thus, one might expect to wait on the order of tens of thousands of generations (or hundreds of thousands of years) for a mutation to appear in the population at all. Furthermore, when a mutation first arrives in the population it arrives on only a single chromosome and as a result is likely to be quickly lost to drift. Under neutrality, and ignoring recurrent mutation, the DTWF model is a martingale, which implies that the probability that a variant present in a single haploid ultimately fixes is one over the total number of haploids in the population. As a result, in order to get a mutation to fix at a site, it will on average require a mutation arising at that site a number of times equal to the population size. The waiting time between each of these mutations is on the order of hundreds of thousands of years. Ultimately, this means that at a single position, the timescale of equilibration is on the order of one over the mutation rate. In humans, this would correspond to about a 100 million generations, or a few billion years. Given that life has only been present on Earth for about 4 billion years, and has clearly been rapidly changing, it seems implausible that any position in any genome could possibly be at equilibrium.

The infinite sites approximation avoids this issue of time-scales by approximating the genome as being infinitely long, with each site having an infinitesimally small mutation rate. In this limit, there are infinitely many sites, but only finitely many of them have variants segregating at any time, and as such it does not make sense to consider the probability of a given site segregating, as for any single site that probability is zero. Yet, one can model how many sites are expected to be segregating, or the distribution of allele frequencies conditioned on a site being segregating. Throughout, we will say the *frequency spectrum* to refer to the expected number of sites that have variants at each frequency. In this approximation, mutations only happen at most once at a given site and as a result we can determine which allele is "ancestral" and which allele is "derived," and we can ignore recurrent mutations. One way to view the frequency spectrum is as a vector, ϕ, where $\phi(k)$ is the expected number of sites at which k individuals have the derived allele. We only track the dynamics of segregating sites (because there are always infinitely many nonsegregating sites), and as such, we ignore any derived alleles if and when they reach fixation, so ϕ is $N-1$ dimensional.

Let ϕ_t denote the frequency spectrum at generation t. Alleles already in the population evolve according to DTWF dynamics, and we expect to "inject" $\theta/2$ new singletons each generation, which we can capture by adding $(\theta/2)\mathbf{e}_1$ to the frequency spectrum each generation, where $\mathbf{e}_1$ is the vector with 1 in the first position and zero in all other positions. One can then either add the mutations first, which would correspond to a model where mutation happens during gamete formation, and then the next generation is formed by sampling from those mutated gametes:

$$\phi_{t+1} = \mathbf{M}^{\mathsf{T}}_{1:-1,1:-1}\left(\frac{\theta}{2}\mathbf{e}_1 + \phi_t\right),$$

or one can consider a model where the next generation is formed first, and then mutates prior to being genotyped:

$$\phi_{t+1} = \frac{\theta}{2}\mathbf{e}_1 + \mathbf{M}^{\mathsf{T}}_{1:-1,1:-1}\phi_t,$$

where the subscripts on $\mathbf{M}$ indicate that we are dropping the first and last rows and columns of $\mathbf{M}$ (corresponding to the nonsegregating states). In the diffusion limit, generations happen infinitely fast so these two models are equivalent, but in the DTWF model these two mutation models are subtly different. In particular, mutating after demographic sampling produces substantially more singletons *in the population*—it is possible to show that the equilibrium of the model with mutation after demographic sampling will have exactly $\theta/2$ more singletons in the population than the model with mutation before demographic sampling. Since the number of singletons in the equilibrium DTWF model where mutation happens after demographic sampling is $\approx 1.12\times\theta$ (Wakeley and Takahashi 2003), this results in a nearly factor of two difference between the models in terms of singletons. This difference is largely washed out in small subsamples of the population but becomes apparent as the sample size gets large. In humans there is some biological evidence for both of these models—siblings can share mutations that are not present in either parent, consistent with mutation in the parental germline, but there is also growing evidence that the first few replications after zygote formation are particularly error prone, which would be consistent with the second model (Gao *et al.* 2016, 2019; Sasani *et al.* 2019). Reality likely involves some combination of these models, indicating that the exact singleton count in large samples is not reliable, as it depends on extremely fine-scale aspects of the underlying model that are not currently well understood. Yet, these details do not appear at all in the Wright–Fisher diffusion, providing further evidence that the diffusion stops providing a good approximation to reality as the sample size gets large.

Note that in either formulation, since we are dropping the first and last columns of $\mathbf{M}$, we lose mass from ϕ each generation, corresponding exactly to those variants that have either reached fixation or been lost by drift. This loss of mass will at equilibrium be offset by the influx of mass from the $(\theta/2)\mathbf{e}_1$ term corresponding to new mutations.

Since the above formulation is written entirely in terms of matrix–vector products, it is then easy to perform rapid approximate calculations using our tricks by simply replacing $\mathbf{M}$ with $\widetilde{\mathbf{M}}$.

Note that these formulations implicitly assume that the number of sites where a new mutation arises each generation is deterministic. A more realistic formulation would have a Poisson number of new mutations—any given mutation is extremely unlikely, but there are many potential mutations throughout the genome, and so a "law of rare events" type argument gives the Poisson distribution. Yet, the variance of the Poisson equals the mean, so the average distance between an observation and the mean is about the square root of the mean. In our setting, that means that if the expected number of mutations per generation is large, then the Poissonian noise about that mean is comparatively small. For example, returning to the example of humans with an effective size of ≈10,000, a mutation rate of $\approx 10^{-8}$, and a genome size of $\approx 10^9$, we expect to see about $10{,}000 \times 10^{-8} \times 10^9 = 10^5$ newly mutated sites per generation. Meanwhile, we expect the fluctuations in the number of mutations to be on the order of ≈1% of the expected number of mutations, showing that this deterministic approximation is quite accurate.

Conditioning on nonfixation

The infinite sites model has some conceptual downsides. For example, methylated CpG sites have extremely high mutation rates (Jónsson *et al.* 2017; Karczewski *et al.* 2020), making recurrent mutation an empirically nonnegligible force at realistic population sizes (Harpak *et al.* 2016). Yet, the infinite sites model rules out recurrent mutation—if the mutation rate is high enough for a given site to mutate twice, then there must be infinitely many mutations in the genome. Furthermore, the notion of an "ancestral" and "derived" allele becomes unclear in the presence of recurrent mutation; if mutations can happen repeatedly at a given site, then the derived allele could previously have gone to fixation and the ancestral allele could be subsequently reintroduced. A related conceptual issue is in the probability of a site being nonsegregating. Under the infinite sites model, any given site is nonsegregating with probability one, and the same holds for any finite number of sites. This prevents the infinite sites model from using the absence of mutations as an indication of natural selection, which has proven to be a powerful technique for measuring gene constraint (Lek *et al.* 2016; Karczewski *et al.* 2020). In particular, while models based on the infinite sites assumption can implicitly make use of nonsegregating sites by tracking the total number of mutations, they run into conceptual issues when, for example, defining the likelihood that a single monomorphic site is experiencing a given level of selection.

An idea that is conceptually similar to the infinite sites model is to allow arbitrary dynamics (e.g. recurrent mutation) but then condition the derived allele on nonfixation. That is, the derived allele is allowed to arise, perhaps even repeatedly at the same site, but we ignore situations in which that derived allele eventually fixes in the population. As a result, if we look at any position in the genome under this model, that position is either nonsegregating with only the ancestral allele present, or it is segregating, but we know that at some point in the past it was nonsegregating and the population was fixed for the ancestral allele. By making this assumption, we may pick a particular allele as being the ancestral allele, and safely know that under this model, the last time the population was monomorphic at this site, it was monomorphic with the ancestral allele. Yet, by allowing noninfinitesimal mutation rates at each position, we can still derive a probability of segregating for each site, and we can incorporate recurrent mutation in a conceptually clean way.

To incorporate conditioning on nonfixation, we can simply perform matrix–vector products as above, but we replace the final row of $\widetilde{\mathbf{M}}$ (corresponding to the derived allele being fixed) with a row of zeros, and we replace the final column of $\widetilde{\mathbf{M}}$ (corresponding to transitioning to having the derived allele being fixed) with a column of zeros. This causes any mass that would have resulted in fixation being removed from the system. As a result, each transition, $\widetilde{\mathbf{M}}^\top \mathbf{v}$, will result in a loss of mass corresponding to the allele trajectories that would have resulted in fixation of the derived allele. After each transition we can then renormalize $\mathbf{v}$ to sum to one. That this is correct follows from noting that we are eliminating all trajectories that result in fixation, and then rescaling all of the remaining probabilities by a constant (the probability of not reaching fixation).

Appendix D: Additional numerical results

Here, we include additional details, plots, and results related to *Impact of mutation, selection, and demography on the DTWF model* section. Throughout *Impact of mutation, selection, and demography on the DTWF model* section, we used slightly modified versions of the demographic histories for the CEU and YRI samples from the 1000 Genomes Project (Consortium 2012) inferred using MSMC (Schiffels and Durbin 2014), as well as a demography for the YRI inferred using Relate (Speidel *et al.* 2019). "YRI" will refer to the MSMC-inferred demography and "YRI (Speidel)" will refer to the Relate-inferred demography.

First, we smoothed out some small fluctuations in sizes older than 100,000 years ago, and assumed that an extremely large population size estimated using YRI but not estimated using CEU was artifactual, opting to use a smaller ancestral size. Second, the present-day YRI population size (36,822) is smaller than the largest sample sizes we wanted to consider, so we added a single generation of 500,000 individuals at present. Years were converted to generation times by dividing by 30 and rounding. The original and modified demographies (ignoring the recent increase in YRI) are presented in Fig. D1.

Modifying the YRI demography this way results in an extreme scenario where many individuals must find a common ancestor one generation ago since a sample size of $n = 300{,}000$ is ≈10× larger than the population size in the previous generation. An interesting consequence of this is that under neutrality there can be many de novo mutations occurring in the most recent generation resulting in singletons, but since the expected number of present-day haplotypes that come from a particular parent in the previous generation is ≈10, variants that are present in more than one but fewer than 10 individuals should be exceedingly rare. As a result, the frequency spectrum under this demography is nonconvex: doubletons are rarer than both singletons and tripletons, but more common than extremely high frequency variants (Fig. D2). This highlights an advantage of our approach in that we can model such unusual frequency spectra whereas the frequency spectrum under the diffusion approximation must be convex for *any* population size history, a result of Sargsyan and Wakeley (2008). We note, however, that this particular result is driven by our simplistic modification of the YRI demography and unlikely to be a good match to actual frequency spectra obtained from large samples of individuals closely related to the YRI sample from the 1000 Genomes Project. Yet, for small sample sizes, we see that the frequency spectra are largely unaffected by this single generation of growth, and the frequency spectra take on more familiar shapes.

To compute the Fisher Information in *Impact of mutation, selection, and demography on the DTWF model* section, we used numerical differentiation on linear interpolations of likelihoods. Specifically, we compute likelihoods by linearly interpolating between precomputed grid points. This method results in slight technical artifacts where the Fisher Information is extremely high on one side of a grid point and extremely low on the other side of the grid point

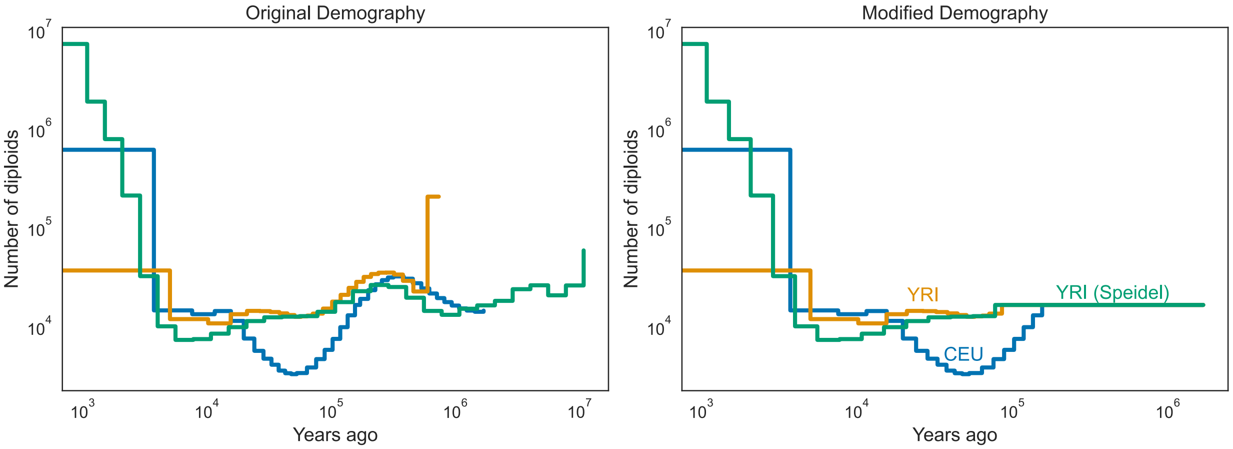

Fig. D1. The original (Schiffels and Durbin 2014) and modified demographies for CEU, YRI, and YRI (Speidel) considered in this paper. Note that in the modified demography, YRI has an additional generation with a population size of 500,000 diploids in the most recent generation that is truncated from the plot.

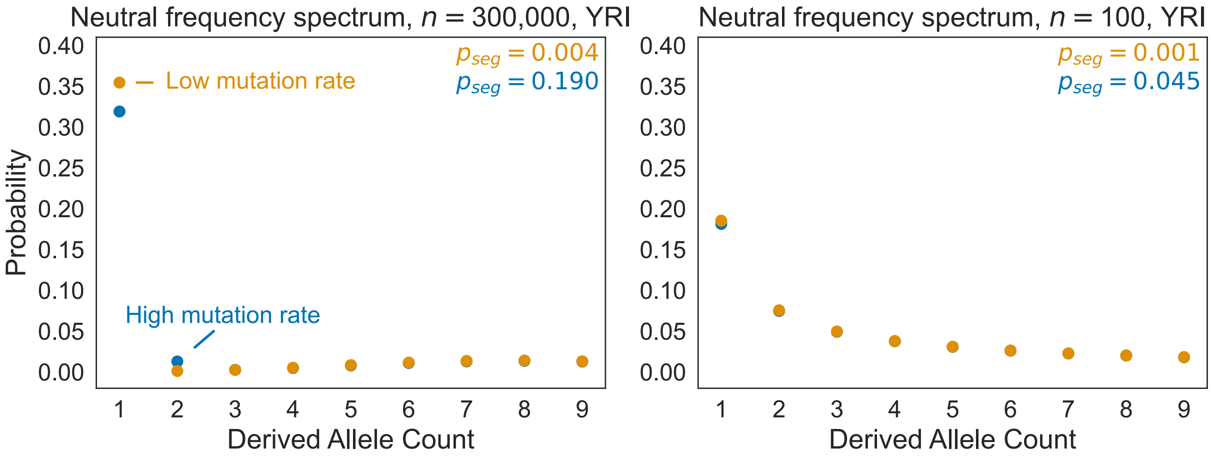

Fig. D2. The lowest entries of the frequency spectra implied by our modified YRI demography for sample sizes of $n = 300{,}000$ diploids (left) or $n = 100$ diploids (right). The high mutation rate corresponds to the mutation rate of a methylated CpG (1.25×10^{-7} per generation) and the low mutation rate roughly corresponds to the rate of transversions (2.44×10^{-9} per generation). The spectra for the two mutation rates almost coincide on the right plot.

with a discontinuity at the grid point. This arises from linear interpolation being nondifferentiable at the grid points. To avoid these technical artifacts, we computed the Fisher Information on a dense grid of points and then smoothed the resulting values using `gaussian_filter1d` from `scipy` (Virtanen *et al.* 2020) with a kernel chosen to visually smooth out the artifactual fluctuations in the Fisher Information.

Appendix E: Convergence of moments

In this section, we consider the moments of the DTWF process as well as our approximation. While we showed in Appendix A that the processes are close in terms of total variation distance, total variation distance can be either overly strict or overly lax in some situations. This arises because total variation is agnostic to any metric on the space of outcomes. Our approximation produces a small difference in total variation distance, but there are other approximations that also have small total variation distance, but result in pathological behavior. For example, consider the process obtained by flipping a coin that comes up heads with probability ε, and if the coin is heads, then in the next generation the frequency of the A allele is zero regardless of its current frequency. If the coin is tails, then the next generation is obtained via the standard DTWF process. It is easy to see that the transition density of this strange process has small total variation distance to the transition density of the usual DTWF process. Yet, this construction has totally outlandish behavior—if we consider a DTWF model without recurrent mutation, and say that the current allele frequency is 1, then under the true DTWF model, the population will be forever stuck at a frequency of 1. On the other hand, the strange construction will eventually (after about $1/\varepsilon$ generations on average) crash to a frequency of 0. This example highlights that being close in total variation is not necessarily sufficient for one process to be a sensible approximation of the other. As such, the remainder of this section will work toward showing that the mean and variance of the transition density of our approximate process are close to the transition density of the full DTWF process.

There are two pieces to our approximation—combining rows of the transition matrix that correspond to Binomial distributions with similar success probabilities, and sparsifying the row—and both affect the moments.

The effect of combining similar rows is straightforward to analyze. Since the mean of a $\mathrm{B}_{N,p}$ distribution is Np, and, assuming that $p \leq 1/2$, we combine rows a row with success probability p with another that differs by at most $c_\varepsilon\sqrt{p/N}$, we can see that this affects the mean by $O(\sqrt{Np})$. In practice, we also choose our representative success probabilities so that while some rows have their means increased by as much as $O(\sqrt{Np})$, those are balanced by rows that have their means decreased by a corresponding amount. As a result, if you pick a row randomly (with probability proportional to the probability of observing that frequency in the previous generation) the difference between its approximate and true means is 0 on average.

Likewise, the variance of $B_{N,p}$ is $Np(1-p)$, which is also altered by something that is $O(\sqrt{Np})$. In both cases, we see that the effect of the perturbation is a lower order term.

Analyzing how the sparsification affects the moments is substantially more technical, but we include it here for completeness. We will prove the closeness of the first two moments of the truncated and nontruncated Binomial distributions by showing that the truncated Binomial distribution is very close to a truncated Normal distribution, for which we can readily compute moments, and then show that those moments are close to those of the Binomial distribution. Our proof relies on a nonuniform version of the celebrated Berry–Esseen theorem (Nagaev 1965), which we include here without proof.

Lemma E1. (Nonuniform Berry–Esseen bound for Binomial, adapted from Nagaev (1965, Theorem 3))**.** Let $X \sim B_{N,p}$, and let $Z \sim \mathcal{N}(0,1)$. Then, there exists a universal constant c such that

$$\left|\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z\right) - \mathbb{P}(Z \geq z)\right| \leq \frac{c}{(1+|z|^3)\sqrt{Np(1-p)}}.$$

Using the Berry–Esseen bound we can show that Binomial distributions truncated as described in Lemma A6 (and used in our approach) are quantitatively similar in distribution to a truncated Gaussian.

Lemma E2. *(Nonuniform Berry–Esseen bound for truncated Binomial). Let $\tilde{X} \sim \tilde{B}^{\varepsilon}_{N,p}$ be a random variable drawn from a Binomial distribution truncated as described in Lemma A6. Define $z_{\min}$ as*

$$z_{\min} := -\sqrt{\frac{\log 2/\varepsilon)}{2p(1-p)}}$$

and $z_{\max}$ as

$$z_{\max} := \sqrt{\frac{\log(2/\varepsilon)}{2p(1-p)}}.$$

Let $\tilde{Z}$ be distributed according to a truncated standard Normal distribution, truncated at $z_{\min}$ and $z_{\max}$. Then,

$$\left|\mathbb{P}\left(\frac{\tilde{X}-Np}{\sqrt{Np(1-p)}} \geq z\right) - \mathbb{P}\left(\tilde{Z} \geq z\right)\right| \leq O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right).$$

Proof. First, note that by construction $\tilde{X}$ and $\tilde{Z}$ are truncated at the same points (that is $z_{\min}$ and $z_{\max}$ are simply the centered and scaled versions of $k_{\min}$ and $k_{\max}$ appearing in Lemma A6) so for any z lying outside of these truncation points, the bound in the Lemma is vacuously true, as the difference in the distributions is 0.

Now, consider a $z \in [z_{\min}, z_{\max}]$. We begin by rewriting the distribution of $\tilde{X}$ in terms of the distribution of a Binomial random variable, $X \sim B_{N,p}$:

$$\mathbb{P}\left(\frac{\tilde{X}-Np}{\sqrt{Np(1-p)}} < z\right) = \frac{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}$$

and using Lemma E1 we can write the right-hand side as

$$= \frac{\mathbb{P}(Z \geq z) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} + O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right),$$

where we used that by Lemma A6 the denominator is at least some constant independent of N, p, and z. We now write the probability involving Z in terms of $\tilde{Z}$

$$\mathbb{P}(Z \geq z) = [\mathbb{P}(Z \geq z_{\min}) - \mathbb{P}(Z \geq z_{\max})]\mathbb{P}(\tilde{Z} < z) + \mathbb{P}(Z \geq z_{\max}).$$

Plugging this result in, we obtain

$$\begin{aligned}\mathbb{P}\left(\frac{\tilde{X}-Np}{\sqrt{Np(1-p)}} < z\right) &= \left(\frac{\mathbb{P}(\tilde{Z} < z)(\mathbb{P}(Z \geq z_{\min}) - \mathbb{P}(Z \geq z_{\max}))}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}\right) \\ &+ \frac{\mathbb{P}(Z \geq z_{\max}) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} \\ &+ O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right).\end{aligned} \tag{A4}$$

We now tackle the two terms on the right-hand side of the previous equation, starting with the second term. Bounding the denominator by a constant, as above, and applying Lemma E1 we see

$$\frac{\mathbb{P}(Z \geq z_{\max}) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} = O\left(\frac{1}{(1+|z_{\max}|^3)\sqrt{Np(1-p)}}\right).$$

Now, noticing that since $z \in [z_{\min}, z_{\max}]$, we have that $|z| \leq |z_{\max}|$ by construction, and so we can loosen this bound to

$$= O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right).$$

We now turn to the first term on the right-hand side of Equation (4), where we use similar tricks:

$$\begin{aligned}
&\frac{\mathbb{P}(Z \geq z_{\min}) - \mathbb{P}(Z \geq z_{\max})}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} \\
&= 1 - \frac{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}(Z \geq z_{\min})}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} \\
&\quad - \frac{\mathbb{P}(Z \geq z_{\max}) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)}{\mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\min}\right) - \mathbb{P}\left(\frac{X-Np}{\sqrt{Np(1-p)}} \geq z_{\max}\right)} \\
&= 1 + O\left(\frac{1}{(1+|z_{\min}|^3)\sqrt{Np(1-p)}}\right) \\
&\quad + O\left(\frac{1}{(1+|z_{\max}|^3)\sqrt{Np(1-p)}}\right) \\
&= 1 + O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right).
\end{aligned}$$

Using these results, we may simplify Equation (4) to obtain

$$\begin{aligned}
\mathbb{P}\left(\frac{\widetilde{X}-Np}{\sqrt{Np(1-p)}} < z\right) &= \mathbb{P}(\widetilde{Z} < z) \\
&\quad + O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right).
\end{aligned}$$

□

With our truncated version of the nonuniform Berry–Esseen theorem, we are now ready to prove that our truncation of the Binomial distribution does not substantially alter the first two moments.

Proposition E1. Let $\widetilde{X} \sim \widetilde{B}^{\varepsilon}_{N,p}$. Then,

$$\mathbb{E}[\widetilde{X}] = Np + O(1).$$

Proof. Recall that for any strictly positive random variable,

$$\mathbb{E}[\widetilde{X}] = \int_0^\infty \mathbb{P}(\widetilde{X} \geq x)\,\mathrm{d}x.$$

Centering and scaling, we see

$$\mathbb{E}[\widetilde{X}] = \int_0^\infty \mathbb{P}\left(\frac{\widetilde{X}-Np}{\sqrt{Np(1-p)}} \geq \frac{x-Np}{\sqrt{Np(1-p)}}\right)\mathrm{d}x.$$

Performing a change of variables and applying Lemma E2, we arrive at

$$\begin{aligned}
\mathbb{E}[\widetilde{X}] = \sqrt{Np(1-p)} \int_{-\frac{Np}{\sqrt{Np(1-p)}}}^{\infty} &\mathbb{P}(\widetilde{Z} \geq z) \\
&+ O\left(\frac{1}{(1+|z|^3)\sqrt{Np(1-p)}}\right)\mathrm{d}z,
\end{aligned}$$

where $\widetilde{Z}$ is a truncated Gaussian, truncated at $-\sqrt{\frac{\log(2/\varepsilon)}{2p(1-p)}}$ and $\sqrt{\frac{\log(2/\varepsilon)}{2p(1-p)}}$.

Noting that

$$\int_{-\frac{Np}{\sqrt{Np(1-p)}}}^{\infty} \frac{1}{(1+|z|^3)}\,\mathrm{d}z < \int_{-\infty}^{\infty} \frac{1}{(1+|z|^3)}\,\mathrm{d}z = \frac{4\pi}{3\sqrt{3}} = O(1)$$

we obtain

$$\begin{aligned}
\mathbb{E}[\widetilde{X}] &= O(1) + \sqrt{Np(1-p)} \int_{-\frac{Np}{\sqrt{Np(1-p)}}}^{\infty} \mathbb{P}(\widetilde{Z} \geq z)\,\mathrm{d}z \\
&= O(1) + Np + \sqrt{Np(1-p)} \int_{-\frac{Np}{\sqrt{Np(1-p)}}}^{\infty} z\,\mathrm{d}\mathbb{P}(\widetilde{Z} = z),
\end{aligned}$$

where the second equality follows from integrating by parts. Now, note that without loss of generality we can assume that $p \geq 1/2$ (by symmetry of the Binomial). Therefore, if we take N to be large enough, the lower limit of the integral is lower than the lower truncation point of $\widetilde{Z}$ so the integral covers the entire domain of $\widetilde{Z}$. Therefore, for sufficiently large N, the integral is exactly $\mathbb{E}[\widetilde{Z}]$. The mean of a truncated standard Gaussian with symmetric truncation points is 0, completing the proof. □

Proposition E2. Let $\widetilde{X} \sim \widetilde{B}^{\varepsilon}_{N,p}$. Then,

$$\begin{aligned}
\mathrm{Var}(\widetilde{X}) &= (1 + O(\varepsilon\sqrt{\log(1/\varepsilon)}))Np(1-p) \\
&\quad + O(\sqrt{Np(1-p)}).
\end{aligned}$$

Proof. Let μ_B be the mean of $\widetilde{X}$. Then, by Lemma E1,

$$\begin{aligned}
\mathrm{Var}(\widetilde{X}) &= \mathbb{E}[(\widetilde{X}-\mu_B)^2] \\
&= \mathbb{E}[(\widetilde{X}-Np)^2] - (\mu_B - Np)^2 \\
&= \mathbb{E}[(\widetilde{X}-Np)^2] + O(1).
\end{aligned}$$

We now evaluate the expectation on the right-hand side

$$\begin{aligned}
\mathbb{E}[(\widetilde{X}-Np)^2] &= \int_0^\infty \mathbb{P}((\widetilde{X}-Np)^2 \geq x)\,\mathrm{d}x \\
&= \int_0^\infty \mathbb{P}(\widetilde{X}-Np \geq \sqrt{x}) \\
&\quad + \mathbb{P}(\widetilde{X}-Np \leq -\sqrt{x})\,\mathrm{d}x.
\end{aligned}$$

We now perform the change of variables $z = \sqrt{x}/\sqrt{Np(1-p)}$, and use Lemma E2, with $\widetilde{Z}$ being the truncated Gaussian corresponding to $\widetilde{X}$:

$$\begin{aligned}
&\int_0^\infty \mathbb{P}(\widetilde{X}-Np \geq \sqrt{x})\,\mathrm{d}x \\
&= Np(1-p)\int_0^\infty 2z\left(\mathbb{P}(\widetilde{Z} \geq z)\right. \\
&\quad \left. + O\left(\frac{1}{(1+z^3)\sqrt{Np(1-p)}}\right)\right)\mathrm{d}z.
\end{aligned}$$

Noting that

$$\int_0^\infty \frac{z}{(1+z)^3}\,\mathrm{d}z = \frac{2\pi}{3\sqrt{3}} = O(1),$$

we obtain

$$\begin{aligned}\int_0^\infty \mathbb{P}(\widetilde{X} - Np \ge \sqrt{x}) &= O(\sqrt{Np(1-p)}) + Np(1-p)\int_0^\infty 2z\mathbb{P}(\widetilde{Z} \ge z)\,\mathrm{d}z \\ &= O(\sqrt{Np(1-p)}) + Np(1-p)\int_0^\infty \mathbb{P}(\widetilde{Z} \ge \sqrt{y})\,\mathrm{d}y,\end{aligned}$$

where the final line follows from the change of variables $z = \sqrt{y}$. A similar computation shows that

$$\int_0^\infty \mathbb{P}(\widetilde{X} - Np \le -\sqrt{x}) = O(\sqrt{Np(1-p)}) \quad + Np(1-p)\int_0^\infty \mathbb{P}(\widetilde{Z} \le -\sqrt{y})\,\mathrm{d}y.$$

We therefore have that

$$\begin{aligned}\mathbb{E}[(\widetilde{X} - Np)^2] &= O(\sqrt{Np(1-p)}) + Np(1-p)\left(\int_0^\infty \mathbb{P}(\widetilde{Z} \ge \sqrt{y})\,\mathrm{d}y\right. \\ &\quad \left. + \int_0^\infty \mathbb{P}(\widetilde{Z} \le -\sqrt{y})\,\mathrm{d}y\right) \\ &= O(\sqrt{Np(1-p)}) + Np(1-p)\int_0^\infty \mathbb{P}(\widetilde{Z}^2 \ge y)\,\mathrm{d}y \\ &= O(\sqrt{Np(1-p)}) + Np(1-p)\,\mathrm{Var}(\widetilde{Z}),\end{aligned}$$

where the final line follows by noting that the mean of a symmetrically truncated standard Normal is 0. $\widetilde{Z}$ is a standard Normal symmetrically truncated at $\pm\sqrt{\frac{\log(2/\varepsilon)}{2p(1-p)}}$. Letting z_{min} and z_{max} be these truncation points and $\phi(\cdot)$ and $\Phi(\cdot)$ be the probability density function and cumulative density function of the standard Normal, respectively, we have that

$$\begin{aligned}\mathrm{Var}(\widetilde{Z}) &= 1 + \frac{z_{\mathrm{min}}\phi(z_{\mathrm{min}}) - z_{\mathrm{max}}\phi(z_{\mathrm{max}})}{\Phi(z_{\mathrm{max}}) - \Phi(z_{\mathrm{min}})} \\ &= 1 + \frac{2z_{\mathrm{max}}\phi(z_{\mathrm{max}})}{\Phi(z_{\mathrm{max}}) - \Phi(z_{\mathrm{min}})} \\ &= 1 + \frac{\frac{\sqrt{2}z_{\mathrm{max}}}{\sqrt{\pi}}\exp\left\{\frac{-z_{\mathrm{max}}^2}{2}\right\}}{\Phi(z_{\mathrm{max}}) - \Phi(z_{\mathrm{min}})} \\ &= 1 + \frac{O(\varepsilon\sqrt{\log(1/\varepsilon)})}{\Phi(z_{\mathrm{max}}) - \Phi(z_{\mathrm{min}})}.\end{aligned}$$

The denominator can be bounded from below by $1 - O(\varepsilon)$, completing the proof. □

Together, these results show that in addition to being close in total variation distance, our approximate process is also close in terms of the first two moments.

Editor: G. Coop