A continuous analog of run length distributions reflecting accumulated fractionation events

Zhe Yu

Thesis submitted to the Faculty of Graduate and Postdoctoral Studies in partial fulfillment of the requirements for the degree of
Master of Science in Mathematics[1]

Department of Mathematics and Statistics
Faculty of Science
University of Ottawa

[1]The M.Sc. program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Abstract

We propose a new, continuous model of the fractionation process (duplicate gene deletion after polyploidization) on the real line. The aim is to infer how much DNA is deleted at a time, based on segment lengths for alternating deleted (invisible) and undeleted (visible) regions. After deriving a number of analytical results for "one-sided" fractionation, we undertake a series of simulations that help us identify the distribution of segment lengths as a gamma with shape and rate parameters evolving over time. This leads to an inference procedure based on observed length distributions for visible and invisible segments. We suggest extensions of this mathematical and simulation work to biologically realistic discrete models, including two-sided fractionation.

Dedications

This thesis is dedicated to my parent, for their endless love, support and encouragement in my life. It is also dedicated to my grandparents, for their inspiration and unconditional kindness to me.

Acknowledgement

I would first like to express my deepest gratitude to my supervisor, Dr. David Sankoff, for the continuous support of my study and research, for his patience, motivation, and immense knowledge. I would like to thank him for his excellent guidance to choose the right direction and successfully complete my thesis. I would never have been able to finish my thesis without his great support. It is my lucky and honour to work with him during my Master. My sincere gratitude also goes to Dr. Benoit Dionne, who has always found him when needed and been generous in providing help.

Besides my advisor, I would like to thank my fellow colleagues in the laboratory, Chunfang Zheng, Arash Jamshidpey, Mona Meghdari. They were always willing to help and gave their best suggestions whether it is technical or non-technical.

Last but not the least, I would like to thank my family: My mother, Jie Tang, the kindest and sweetest women ever, who was always there cheering me up and stood by me though the good times and bad. My father, Jixiang Yu, the best role model, who was supporting me and encouraging me with his best wish. They gave birth to me at the first place and support me spiritually throughout my life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the course of evolution, new genomes occasionally arise by duplication or triplication of an existing genome, so that there are two or three identical copies called homeologous (not to be confused with homologs), of each maternal and each paternal chromosome. After a (usually) transient period of polyploidy marked by unusual patterns of meiosis where more than just one maternal and paternal chromosome are aligned and recombine, processes of sequence divergence and chromosome rearrangement lead to more familiar diploid patterns. At the same time a process of *fractionation* eliminates some or most of the duplicate genes, some from each chromosomal copy, but in the simplest model, never all members of a duplicate pair or triple - for reasons of viability. Fractionation processes have been surveyed across evolutionarily diverse types of eukaryote organisms [1].

Since one copy of a duplicate pair of genes must be retained, we can identify not only the chromosomal regions that have been retained – by simple observation of the genome – but also each region that is now invisible – by reference to the duplicate chromosome that has necessarily retained a copy of this region. Thus, the

data on which inferences about the deletion process can be made consist of alternating segments of deleted and undeleted genome of varying lengths.

Among the important questions about the nature of the deletion process, we can ask whether deletion proceeds one gene at a time or by larger chromosomal fragments. The eventual inference problem is, given a previously doubled genome, if we can observe segments of this genome where all the genes have been deleted from one chromosome, but not from the homeologous chromosome, alternating with segments where both gene copies have been preserved, what can we deduce of the process by which duplicated genes are deleted? The main complication here is that for an observed segment consisting of several deleted genes this may be the accumulated result of several concatenated or overlapping deletion events. (Of course, every segment containing only non-deleted genes has been affected by zero deletion events.)

In this thesis, we model the process as the deletion of segments from the real line, with a biologically realistic treatment afforded both for the dynamics of the process and overlapping deletions. Previous work focused on the difficult question of how many overlapping deletion events are responsible for each contiguous deleted region [2, 3, 4], but was not able to account analytically for the dynamics of the process.

In this thesis we attack and solve the inference problem of the size, form and spacing of deletion events, allowing for a number of sweeps over the genome as a way of accounting for overlapping deletions. We carry this out in a continuous analog of the original discrete gene-order context, and address the "one-sided" version of the problem, where all deletions occur on one of the duplicate chromosomes.

I will conclude this introduction with a discussion of the background of the topic, and the new approach taken in this thesis. In Chapter 2, I present the new model, and give a number of analytical results. Chapter 3 describes the development of

our simulation package, applicable both to discrete and continuous models. Chapter 4 describes the analysis of an extensive simulation study, leading to a well-justified hypothesis about the distribution of the lengths of individual deleted and undeleted segments of the chromosome after a number of sweeps, and to an inference procedure, based purely on the observation of these distributions, of both the parameters of the deletion process and the number of sweeps. This is the first reported solution of this problem. Chapter 5 contains summary remarks and conclusions.

Most of the material in this thesis has been incorporated in a submission accepted for publication in a special 2016 issue of BMC Bioinformatics for papers to be presented at the 14th Annual RECOMB Workshop on Comparative Genomics in Montreal, October 11-14, 2016.

1.1 Background

There has been a certain amount of work on the quantification of the fractionation process, starting in 2006 with [5], which claimed deletions involved one gene at a time, and [6], which treated the number of genes deleted in a single event as a random variable with mean greater than 1. Other work of this kind includes [1] and [7]. However, the modelling of fractionation where the whole genome evolves as a stochastic process began with [2]. The previously unstudied phenomenon taken into account in that work was the overlap of deletion events, something that assumes much importance soon after the fractionation process commences. Overlap must be handled differently if all deletions occur from one copy of the genome or in either copy. To isolate the most important aspect of overlap, [2] gave analytical results for the case where deletions all occurred on one copy ("one-sided" model). Then [3] extended this

to the more realistic case where deletion could occur at different rates, or the same rate, from either copy of the genome ("two-sided" model). This analysis was more difficult and could not be taken as far as with the one-sided model.

For the one-sided model, a closed form solution of how many deletion events contribute to a deleted region after a single event (i.e., at a single increment in the fractionation process) was obtained in [4].

All these studies modelled the effect of individual deletion events. This inevitably encountered problems due to the rapid loss of chromosome length, and impeded the effective use of simulations. In this thesis, these problems are eliminated by considering deletion as a probabilistically homogeneous process sweeping over the whole positive real line, rather than some finite string of genes. Genome length, which is not relevant to the question of inferring the statistics of local deletion, plays no role, and is even avoidable in our simulations by completely replenishing a chromosome shortened in a sweep by a chromosomal segment having undegone the same sweeps.

Chapter 2

A new model

We model the fractionation process in terms of a number of successive sweeps of a point process with parameter ν on the positive reals, i.e., $\nu \in \mathbf{R}^+$, representing one copy of the genome. At the origin, we say that all points of this genome are "visible". A deletion event, rendering a segment of exponentially (mean μ) distributed length "invisible", occurs at each point determined by the point process. The second copy of the genome remains undisturbed throughout and retains a 1-to-1, length preserving, correspondence with the fractionating copy, without regard to any disruption caused by invisibility. In applications, the acceptance of the one-gene-at-a time theory of deletion depends on whether μ is below or above a certain absolute value, but the present work is part of the mathematical preliminaries to the practical questions. The eventual goal of this work is to determine the relative size of the "spacing" parameter ν and the deletion length parameter μ. The model innovation here is to introduce the parameter ν in the place of a rate parameter in previous work, which was awkward to work with.

During the first sweep, illustrated at the top of Figure 2.1 at time (or step) $t = 1$,

the first *deletion point* x_1 is determined by sampling from the exponential distribution

$$\rho(x) = \frac{1}{\nu} e^{-\frac{x}{\nu}}, \quad x \geq 0, \tag{2.0.1}$$

with mean ν. Then a deletion length a_1 is chosen from another exponential distribution

$$\gamma(a) = \frac{1}{\mu} e^{-\frac{a}{\mu}}, \quad a \geq 0, \tag{2.0.2}$$

with mean μ. Normally, $\nu \gg \mu$, but this is not necessary to the analysis. The segment $[x_1, x_1 + a_1)$ is "deleted", or is designated as invisible. The next deletion point x_2 is chosen by sampling x_2' from the first exponential distribution (mean ν), so that $x_2 = x_2' + x_1 + a_1$. Then the length a_2 of the second deleted segment is determined by sampling from γ again. The process continues in this way to find $x_3, a_3, \ldots$ Concatenating only those segments that are still visible, we see that $x_1, x_2, \ldots$ are points determined by a point process with parameter ν. Associated with each of these points x is an "event counter" $C(x)$. Initially, each $C(x) = 1$. We define a function $\pi_t(i), i = 1, \ldots$ measuring the proportion of event counters registering i events at time $t \geq 1$. Thus $\pi_1(1) = 1$ and $\pi_1(j) = 0$, for all $j > 1$.

At times $t = 1, 2, \ldots$, the second, third, ... sweeps begin, all independent of the first sweep and each other, and each applied to the concatenated visible segments only. We sample $x_1^{(t)}$ and $a_1^{(t)}$ in the same way as x_1 and a_1 according to ρ and γ, respectively, to determine a deletion interval $[x_1^{(t)}, x_1^{(t)} + a_1^{(t)})$.

If the interval $[x_1^{(t)}, x_1^{(t)} + a_1^{(t)})$ contains no previously defined deletion point, a new event counter at $C(x_1^{(t)})$ is set at 1. If $[x_1^{(t)}, x_1^{(t)} + a_1^{(t)})$ already contains $j > 1$ deletion points $z_1, \ldots, z_j$, the event counter at $C(x_1^{(t)})$ is set at $1 + \sum_{i=1}^{j} C(z_i)$. The j deletion points $z_1, \ldots, z_j$ become invisible, along with the rest of the segment $[x_1^{(t)}, x_1^{(t)} + a_1^{(t)})$

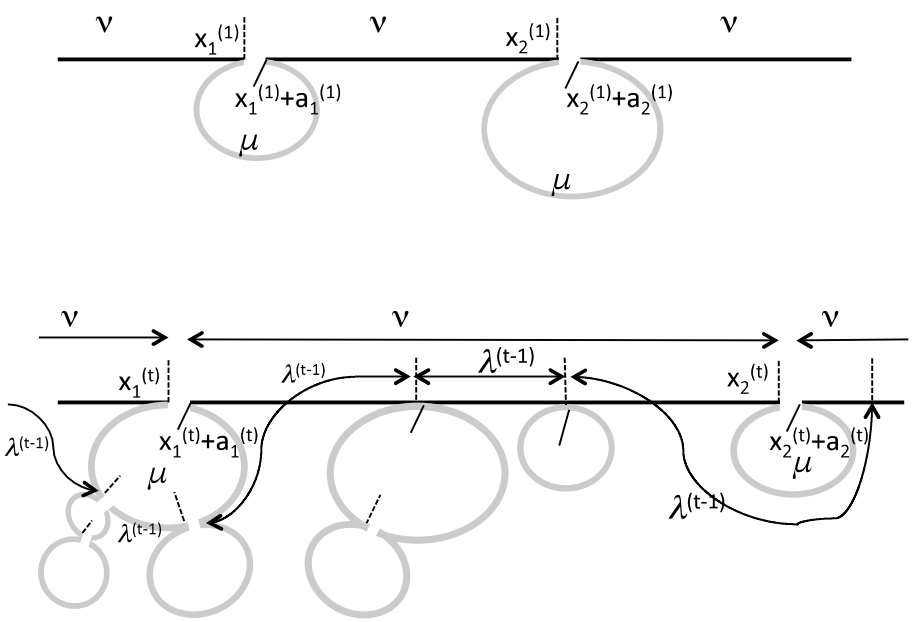

Figure 2.1: Processes pertinent to first sweep and t-th sweep. Solid horizontal bars represent the visible regions of the genome. Grey curves represent invisible regions. Dashed markers represent deletion points, solid markers represent end of deletion segments. ν and μ are the means of the deletion point spacing and deletion segment length variables, while $\lambda^{(t-1)}$ is the mean space ($=\lambda_{t-1}$ in the text) between visible deletion points after the $t-1$-st sweep.

that contains them.

We find the next deletion point by sampling $x_2^{(t)'}$ from ρ, and setting $x_2(t) = x_1^{(t)} + a_1^{(t)} + x_2^{(t)'}$. We continue the t sweep, adding visible deletion points and making others invisible. Some deletion points from the earlier sweep will remain unchanged, i.e. are still visible. The $x_i^{(t)}$ by themselves define a point process with parameter ν on the concatenated visible segments. But the $x_i^{(t)}$ and the additional deletion points remaining from the earlier sweep define a process with mean λ_t, a parameter that decreases with t, as the undeleted segments are interrupted by more and more deletions. This parameter is important as it is directly inferable from the observed

genome at time t.

More important, it is clear, that at each sweep, more and more of the genome becomes invisible. Since each concatenation of visible segments still extends to the positive reals, we cannot observe directly how much the genome has been reduced in absolute terms. But thanks to the length-preserving isomorphism between the second copy of the genome and the fractionating one, for any large finite interval we can observe the proportion of the genome that is left by time t and we can predict that it is approximately $(1 - \frac{\mu}{\nu+\mu})^t$.

We will calculate λ and the number of deletion points in $[x_i, x_{i+1})$, as well as the distribution $p(j), j = 1, ...$ of the number j of pre-existing deletion points in intervals deleted during each sweep, and then discuss how to calculate $\pi_1(j), j \geq 1$, the proportion of event counters with $C = j$.

2.1 The length of undeleted segments λ

After the first sweep, x_i is the only deletion point in $[x_i^{(1)}, a_i^{(1)})$ and the only deletion point in the visible $[x_i^{(1)}, x_{i+1}^{(1)})$, so that $\lambda_1 = \nu$. During the second sweep, the number of these first-sweep deletion points that the visible $[x_i^{(2)}, x_{i+1}^{(2)})$ contains is Poisson distributed with mean $\frac{\nu}{\nu+\mu}$, while the remaining first-sweep deletion points that the invisible $[x_i^{(2)}, a_i^{(2)})$ contains are Poisson distributed with mean $\frac{\mu}{\nu+\mu}$. (These are approximations, since the true means are $\frac{x_{i+1}^{(2)} - x_i^{(2)}}{x_{i+1}^{(2)} + a_i^{(2)} - 2x_i^{(2)}}$ and $\frac{a_i^{(2)} - x_i^{(2)}}{x_{i+1}^{(2)} + a_i^{(2)} - 2x_i^{(2)}}$, respectively.) In addition the visible segment contains one new deletion point, created during the second sweep itself. We can then predict λ_2 to be roughly

$$\hat{\lambda}_2 = \frac{\nu}{1 + \frac{\nu}{\nu+\mu}}. \tag{2.1.1}$$

Suppose λ_{t-1} is the parameter of the point process that generates the deletion points visible after sweep $t-1$. Then, in the sweep at time t, the number of deletion points that the invisible $[x_i^{(2)}, a_i^{(2)})$ will contain is Poisson distributed with mean $\frac{\mu}{\lambda_{t-1}}$. The number of deletion points in the visible $[x_i, x_{i+1})$, not including x_i, is Poisson distributed with mean $\frac{\nu}{\lambda_{t-1}}$. In addition, the visible segment contains one new deletion point, created during the t-th sweep itself. λ_t can thus be predicted to be approximately

$$\hat{\lambda}_t = \frac{\nu}{1 + \frac{\nu}{\hat{\lambda}_{t-1}}}. \tag{2.1.2}$$

Since $\hat{\lambda}_1 = \nu$,

$$\hat{\lambda}_t = \frac{\nu}{t}. \tag{2.1.3}$$

2.2 The treatment of overlapping deletions.

The discussions in this section and the next do not depend on t, so let Λ be the exponential distribution with mean λ. From [4], the proportion of undeleted regions accounted for by segments of length ldl is $\frac{l\Lambda(l)}{\lambda}dl$, where $\lambda = \int_0^\infty l\Lambda(l)dl$. Then the probability p_0 that a deletion event contains no extant deletion points is

$$p_0 = \int_{l=0}^{\infty} \frac{l\Lambda(l)}{\lambda} \int_{x=0}^{l} \frac{1}{l} \int_{y=0}^{l-x} \gamma(y)dy\,dx\,dl. \tag{2.2.1}$$

Carrying out the integrations, we find

$$p_0 = \frac{\lambda}{\mu + \lambda}. \tag{2.2.2}$$

The probability p_1 that a deletion event overlaps exactly one existing run of

deletions is:

$$p_1 = \frac{1}{\lambda}\int_{l=0}^{\infty}\int_{z=0}^{\infty}\Lambda(l)\Lambda(z)\int_{x=0}^{l}\int_{y=l-x}^{l-x+z}\gamma(y)dy\,dx\,dz\,dl \tag{2.2.3}$$

$$= \frac{\lambda}{\mu+\lambda}\cdot\frac{\mu}{\mu+\lambda}. \tag{2.2.4}$$

It can be proved by induction that the probability a deletion event overlaps exactly q existing runs of deletions is:

$$p_q = \frac{\lambda}{\mu+\lambda}\left(\frac{\mu}{\mu+\lambda}\right)^q. \tag{2.2.5}$$

Thus we have the surprisingly uncomplicated result that the number q of pre-existing runs of single-copy regions overlapped by a new deletion event is geometrically distributed on $q = 0, 1, \ldots$ with parameter $\mu/(\mu+\lambda)$.

2.3 The distribution of event counts π.

The event count $C(x)$ at a visible deletion point x tells us how many deletion events have occurred to make up the invisible segment adjacent to x. In contrast to the undeleted segments, where we know that no events occurred, observing that a segment has been deleted does not tell us $C(x)$. Some work has focused on the distribution $\pi(i)$ of the probabilities that a deletion point x has $C(x) = i$, and we are able to calculate how π changes with each sweep. Then we can update π_t by a linear combination of the distribution of changes due to the deletion and the existing π_{t-1}. Let $\Delta(i)$ represent the change in π_i at any sweep t. This can be calculated from equation (2.2.5) and the net effect that a deletion overlapping q existing runs has on the various π. Calculation

details are presented in Appendix A,

$$\Delta(1) = p_0 - p_1\Big[\pi(1)\Big] - 2p_2\Big[\pi(1)\Big] - 3p_3\Big[\pi(1)\Big] - 4p_4\Big[\pi(1)\Big] - ... \tag{2.3.1}$$

$$\Delta(2) = p_1\Big[\pi(1)\Big] - p_1\Big[\pi(2)\Big] - 2p_2\Big[\pi(2)\Big] - 3p_3\Big[\pi(2)\Big] - 4p_4\Big[\pi(2)\Big] - ... \tag{2.3.2}$$

$$\Delta(3) = p_1\Big[\pi(2)\Big] + p_2\Big[\big(\pi(1)\big)^2\Big] - p_1\Big[\pi(3)\Big] - 2p_2\Big[\pi(3)\Big] - 3p_3\Big[\pi(3)\Big] - 4p_4\Big[\pi(3)\Big] - ... \tag{2.3.3}$$

$$\Delta(4) = p_1\Big[\pi(3)\Big] + 2p_2\Big[\pi(1)\pi(2)\Big] + p_3\Big[\big(\pi(1)\big)^3\Big] \tag{2.3.4}$$

$$- p_1\Big[\pi(4)\Big] - 2p_2\Big[\pi(4)\Big] - 3p_3\Big[\pi(4)\Big] - 4p_4\Big[\pi(4)\Big] - ...$$

$$\Delta(5) = p_1\Big[\pi(4)\Big] + p_2\Big[2\pi(1)\pi(3) + \big(\pi(2)\big)^2\Big] + 3p_3\Big[\big(\pi(1)\big)^2\pi(2)\Big] + p_4\Big[\big(\pi(1)\big)^4\Big] \tag{2.3.5}$$

$$- p_1\Big[\pi(5)\Big] - 2p_2\Big[\pi(5)\Big] - 3p_3\Big[\pi(5)\Big] - 4p_4\Big[\pi(5)\Big] - 5p_5\Big[\pi(5)\Big] - ... \tag{2.3.6}$$

...

Unfortunately, even knowing the dynamics of C does not help us with the inference problem, since the number of events associated with an invisible segment, is not directly associated with the total length of the segment. It is known that the overlapping gamma variables making up each segment are related in a complex way, and cannot simply be treated as the sum of gammas drawn a single population.

This leads us to the approach in the next two sections, where simulations strongly suggest the functional form of the distribution of invisible segment lengths, including shape and rate parameters that can be observed, leading to inference of the simulation parameters based on the observations.

Chapter 3

Simulation

The most successful previous attack on the identification of the deletion parameter appears in [4]. Here, the focus was on the classic discrete problem of whether the geometric mean parameter is exactly 1, or greater than 1. Biologically, this is equivalent to deciding between a functional (the loss of a single gene being due to its redundance) and a structural explanation (the loss of a few genes at a time, basically a random chunk of DNA, increasing overall efficiency of the genome).

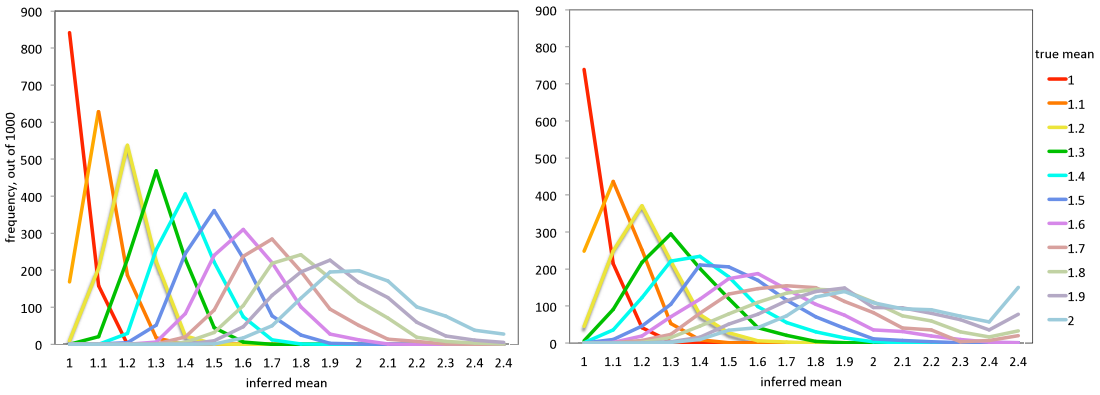

Figure 3.1: Frequency of $\hat{\mu}$, the value for which $D^{(S_i)}_{\mu,N,1-\theta}$ between the sample cumulative and the distribution $F_{\mu,N,1-\theta}$ is minimal. All data involve a proportion of $1-\theta = 0.20$ deleted genes. Left: $N = 900$, right: $N = 300$.

When deletion events remove a number X of contiguous genes, where X is drawn from a geometric distribution with mean μ, the debate may be formulated as a distrimination test of whether $\mu = 1$ or $\mu > 1$. In [4], through 1000 simulations with various combination of the parameters μ, N and $1-\theta$, the distribution $F(\mu)$ of gap length for each μ was calculated accurately shown in Figure 3.1. Starting with N pairs of genes, deletion events for each μ were simulated until $1-\theta$ proportion of genes had been deleted. Then the distribution of gap length in each one run was fit by the Kolmogorov-Smirnov statistic $D^{(S_i)}_{\mu,N,1-\theta}$ for each fifteen values of μ and assigned to $\hat{\mu}$ with minimal value of D. As at the left of Figure 3.1, the orange line shows that there were around 620 out of 1000 simulations that gave a gap size distribution closest to the previously calculated $F_{1.1,900,0.2}$; Around 190 fit best with $F_{1,900,0.2}$, and 190 fit $F_{1.2,900,0.2}$. Similarly, the separate curves represent simulations with different $\mu = 1, 1.1, 1.2, ..., 2$, showing what $\hat{\mu}$ is inferred. In this discrete model, we see that if θ is not too big, and N is big enough, the estimation is fairly accurate.

This thesis has the more ambitious goal of inferring both the parameters of the deletion event, and, in the process, the time parameter (number of sweeps). I built a suite of algorithms to simulate the model in Chapter 4, both the continuous version (non-discretized, aside from that imposed by the computer hardware), and a discrete version of the same model. The code for the algorithms is included in Appendix B. I have not done yet any systematic experimentation with the discrete version, focusing on the continuous model of Chapter 2.

The deletion events occur on visible segments, which initially cover some very long interval $[0, G]$, large enough to minimize "edge" effects. Starting at 0, lengths a_i, sampled from an exponential distribution with mean μ, are deleted one-by-one, separated by lengths x_i, sampled from another exponential distribution with mean ν.

This procedure terminates at G, and the remaining visible segments are concatenated, though bookkeeping is maintained on where the deletions occurred and how large they were. The total length Σa_i deleted is then replenished by adding visible segments to the end from a replicate trial, before the next sweep begins.

The detailed descriptions and computation algorithms are indicated in Algorithm 4 and Algorithm 5 in Appendix B. Our simulation experiments were based on initial visible segments of length $G = 10,000$, which is very long in comparison to the deletion lengths with $\mu \leq 10$. In other words, we do not risk artificial effects, like a disappearing genome, after a few sweeps, $t \leq 10$. Moreover, after each sweep, if the total undeleted length $= L$, we add in segments of length totaling $10,000 - L$, copied from a replicate trial. The program, written in Java, was repeated 5 times for each configuration of the parameters μ, ν and t. Each set of 5 trials averaged a total of less than 3 minutes on a Lenovo Y50 laptop.

After t sweeps, we calculated the distribution of segment lengths for both the invisible and visible parts of the model genome.

Chapter 4

Parameter Estimation

After carrying out large numbers of simulations, I systematically used Cullen-Frey graphs to select the most appropriate distributions. As detailed below, the results were the same over the whole range of simulation parameter values. The distribution parameters could then be estimated by Maximum Likelihood Estimation. Probability histograms, CDF, Q-Q, and P-P plots were then examine to confirm the goodness-of-fit between the empirical distributions and the parametric distributions.

4.1 Equations

The results of the simulations strongly suggest that the lengths of the invisible segments are gamma distributed, as illustrated in the Cullen-Frey graphs at the top of Figure 4.1. Since skewness and kurtosis are not robust, in order to take into account the uncertainty of kurtosis and skewness values from empirical data, we performed 1000 bootstrap samples, which are constructed by random sampling with replacement from the original data set, and plotted them on Cullen-Frey graph. As the parameters ν, μ and t change, the moments of the simulated distributions also change, but

remain those of a gamma distribution. Similarly, the distribution of the lengths of the visible segments is always exponential, as at the bottom of Figure 4.1, with rate

$$\lambda^{-1} = \frac{t}{\nu}. \tag{4.1.1}$$

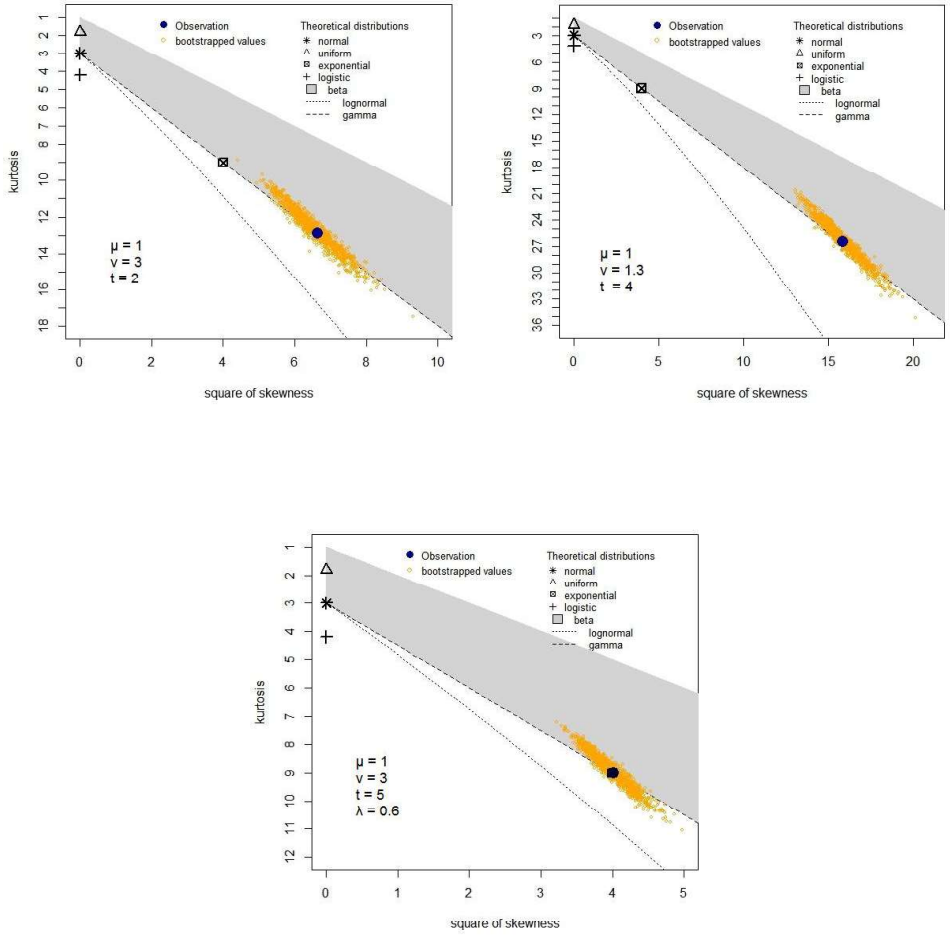

Figure 4.1: Cullen-Frey diagrams for length distributions of invisible (top) and visible (bottom) segments.

As a first step towards the ability to infer μ and ν from the length distributions

of invisible and visible segments, we would like to predict α and β, the shape and rate parameters of the gamma distribution, from t, μ and ν. Table 4.1 suggests, for a fixed value of t and a fixed value $\frac{\mu}{\nu}$, that shape is constant as μ changes, and that the rate is inversely proportion to μ.

Table 4.1: Simulated values of shape and rate when $\frac{\mu}{\nu} = \frac{1}{3}$, for a range of values of μ, and $t = 2$.

μ	ν	shape α	rate β	$1/\beta$
1	3	0.8995	0.7802	1.2818
2	6	0.8713	0.3646	2.7428
3	9	0.8943	0.2558	3.9098
4	12	0.8552	0.1864	5.3656
5	15	0.8480	0.1504	6.6471
6	18	0.8673	0.1251	7.9956
7	21	0.8793	0.1045	9.5728
8	24	0.9191	0.0961	10.4090
9	27	0.9150	0.0882	11.3406
10	30	0.8293	0.0748	13.3631

Similar results hold for each combination of t and $\frac{\mu}{\nu}$, with different shape constants and rate proportions. Figure 4.2 shows how the shape constant varies with t for four values of $\frac{\mu}{\nu}$.

The four coefficients of the linear relationships inferred from Figure 4.2 are plotted in Figure 4.3. Fitting this curve with a quadratic yields

$$\alpha^{-1} - 1 = [-0.1725(\frac{\mu}{\nu})^2 + 0.5333\frac{\mu}{\nu} - 0.039](t-1). \tag{4.1.2}$$

As for the rate parameter of the gamma, Figure 4.4 shows that it is the logarithm of the rate that behaves linearly over time for a fixed value of $\frac{\mu}{\nu}$.

The four coefficients of the linear relationships inferred from Figure 4.4 are plot-

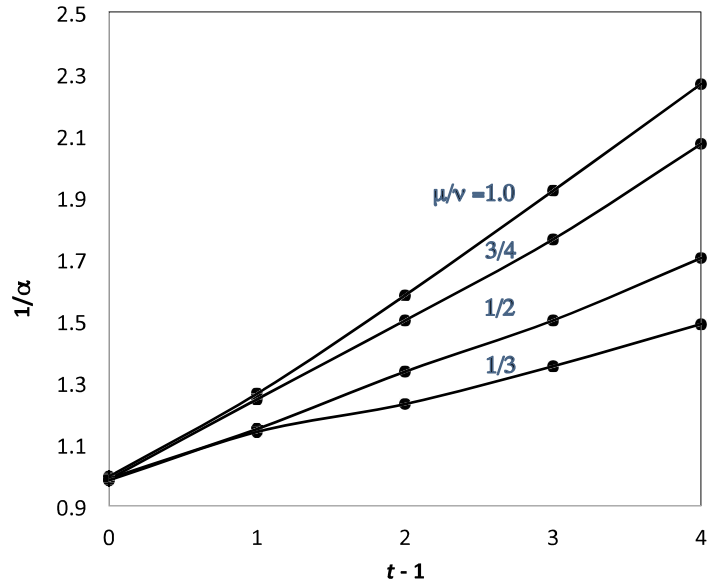

Figure 4.2: Linear relation between $1/\alpha$ and $t-1$ for fixed $\frac{\mu}{\nu}$.

ted in Figure 4.5. Fitting this curve with a quadratic yields

$$\beta^{-1} = \mu \exp[(-0.2458(\frac{\mu}{\nu})^2 + 0.9257\frac{\mu}{\nu} - 0.0212)(t-1)] \tag{4.1.3}$$

Figure 4.6 shows one example of the fit of the Gamma and Exponential distributions to deleted and undeleted seqments, respectively. The density and CDF plots both show a very good fit to the corresponding distributions. The Q-Q and P-P plots are good for detecting lack-of-fit at the distribution tails and centre, respectively. In the example, except some outliers at the upper tail, the fitted distributions are a good fit to the empirical distributions. The deviation at the tail is more ensued from details of the simulations (G not large enough, replenishment of deleted segments) than from the model.

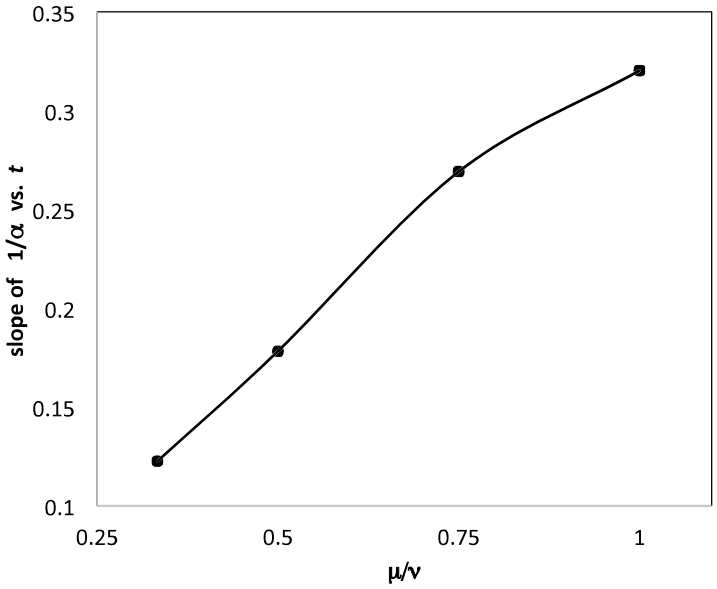

Figure 4.3: Relation between slope of $1/\alpha$ as a function of t, and $\frac{\mu}{\nu}$.

4.2 Methods

The observable quantities in our model are the distribution of visible segment lengths, predicted to be exponential with mean λ, and the shape and rate parameters α and β of the predicted gamma distribution of invisible segment lengths. These three observable quantities are related to the unknown model parameters μ, ν, and t through equations (4.1.1), (4.1.2) and (4.1.3). With the given value of these parameters, we can estimate the values of μ, ν, and t.

Lacking a closed form solution for μ, ν, and t in terms of λ, α and β, we use the following procedure. Since t must be an integer, we can find values of ν_t and μ_t for each $t = 1, 2, \ldots$ with equations (4.1.1) and (4.1.2). Then we can solve equation (4.1.3) to find β_t.

We then compare all the β_t, for $t = 1, 2, \ldots$ with the β observed in the simulation,

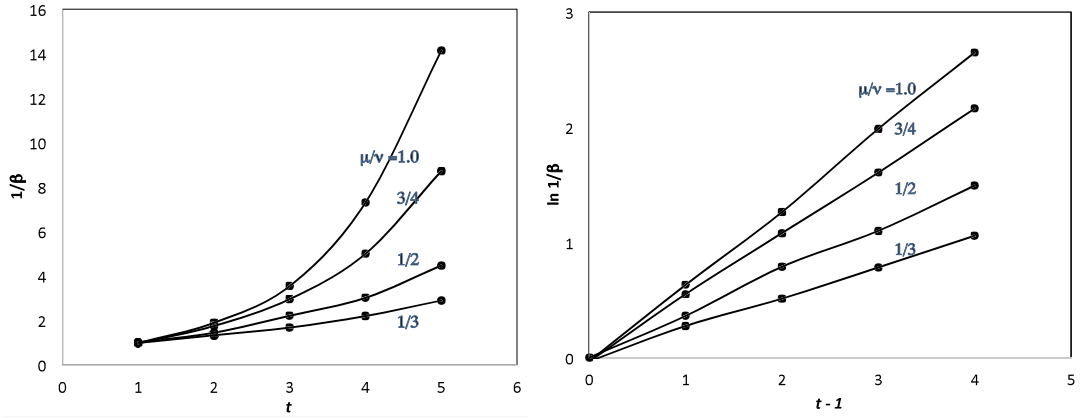

Figure 4.4: Relation between 1/rate $(1/\beta)$ as a function of t for fixed $\frac{\mu}{\nu}$.

and set

$$\hat{t} = \arg\min \{\frac{\beta - \beta_t}{\beta}\} \tag{4.2.1}$$

4.3 Examples

As an example, in one set of simulations where $\mu = 1$, $\nu = 3$ and $t = 5$, the experimental value of parameters are $\lambda^{-1} = 1.665595$, $\alpha = 0.6711252$ and $\beta = 0.3504422$. When $t \leq 2$, there is no solution for μ. For $t > 2$, Table 4.2 shows the results of this procedure, where 100δ is 100 $\times$ the normalized difference between β and β_t in equation (4.2.1).

Table 4.2: Example with $\mu = 1$, $\nu = 3$, $t = 5$, $\lambda^{-1} = 0.16656$, $\alpha = 0.6711$, $\beta = 0.3504$. Framed entry indicates the t most consistent with the observed data on α, β and λ.

par\time t	3	4	5	6	7
μ	1.2317	1.0635	1.0216	1.0186	1.0331
ν	1.8012	2.4015	3.0019	3.6023	4.2027
β_t	0.3006	0.3385	0.3387	0.3253	0.3068
100δ	14.2340	3.4092	$\boxed{3.3636}$	7.1704	12.4566

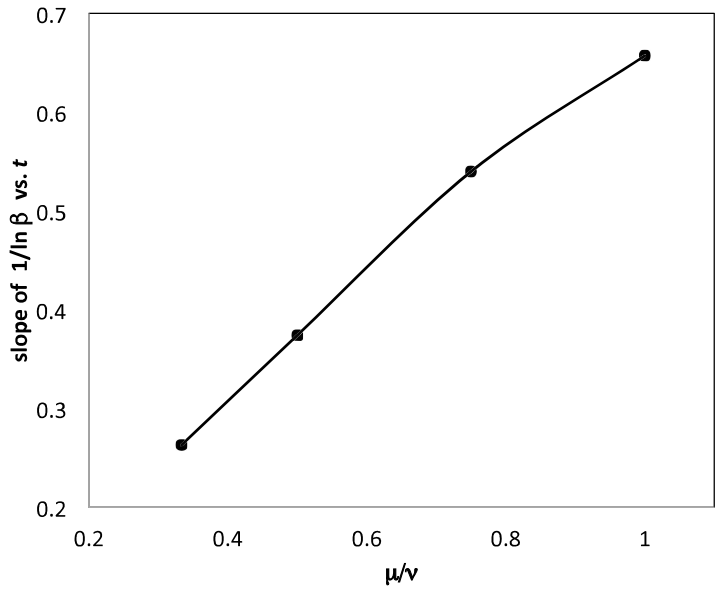

Figure 4.5: Relation between slope of $\ln 1/\beta$ as a function of t, and $\frac{\mu}{\nu}$.

The minimum value of 100δ occurs when $t = 5$, expressing the fact that the inferred values of μ and ν, together with $t = 5$, are the parameter values most consistent with the observed values of α, β and λ. Other typical examples spanning a range of parameter values are given in Tables 4.3, 4.4 and 4.5.

Table 4.3: Example with $\mu = 6$, $\nu = 12$, $t = 2$, $\lambda^{-1} = 0.17$, $\alpha = 0.8488$, $\beta = 0.12063$.

par\time t	2	3	4	5	6	7
μ	5.7892	4.7235	4.7286	4.9648	5.2990	5.6560
ν	12	18	24	30	36	42
β_t	0.1195	0.1406	0.1342	0.1220	0.1094	0.0976
100δ	0.9107	16.5215	11.2325	1.1655	9.3410	19.0791

It can be seen, at least in these diverse examples, that the inference procedure generally identifies the correct value of t, and good estimates of μ and ν.

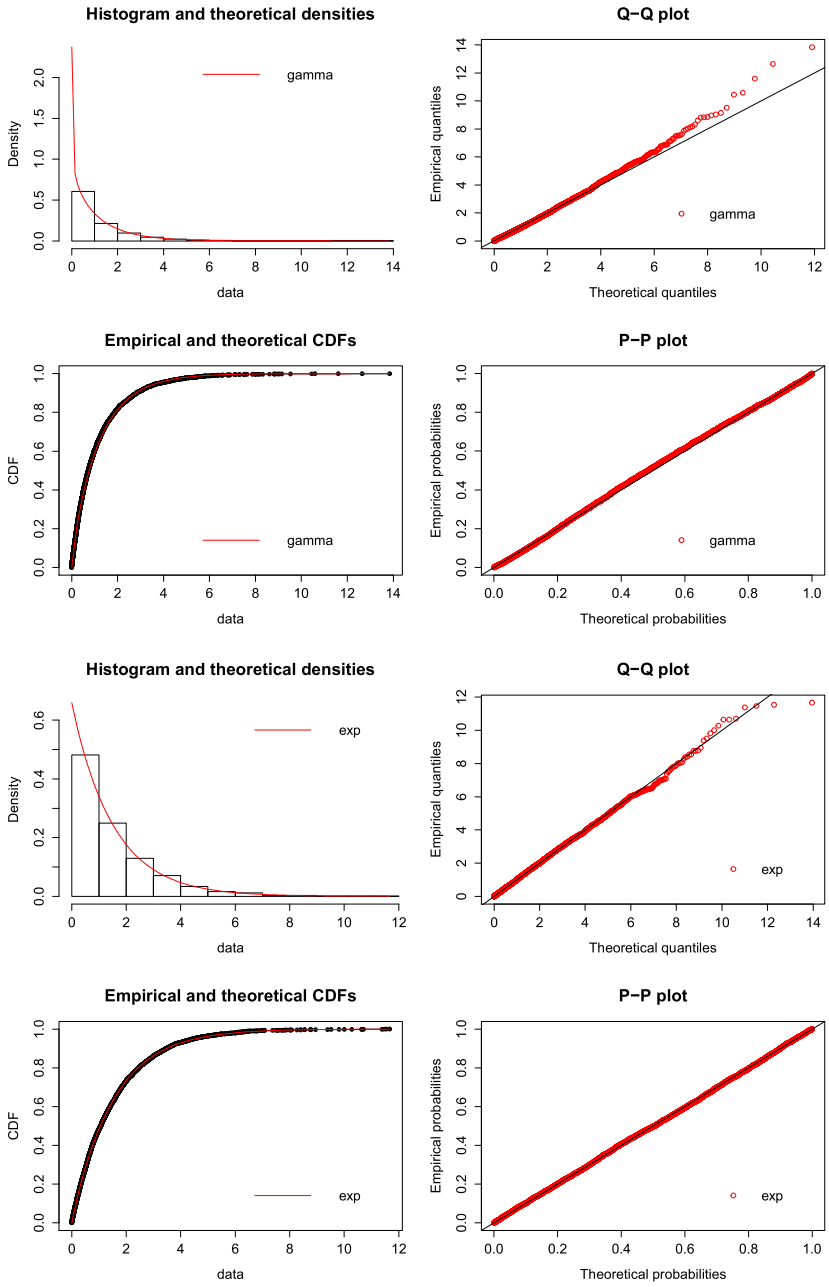

Figure 4.6: Four Goodness-of-fit plots for predicted distribution fitted to invisible and visible sequences, which starting with $\mu = 1, \nu = 3, t = 2$. Top: invisible, Bottom: visible.

Table 4.4: Example with $\mu = 1$, $\nu = 3$, $t = 3$, $\lambda^{-1} = 1.017737$, $\alpha = 0.7977859$, $\beta = 0.5649623$

par\time t	2	3	4	5	6
μ	1.4006	1.0332	0.9909	1.0102	1.0523
ν	1.9651	2.9477	3.9303	4.9129	5.8954
β_t	0.4271	0.5606	0.5596	0.5246	0.4810
100δ	24.39	$\boxed{0.7780}$	0.9497	7.15	14.87

Table 4.5: Example with $\mu = 5$, $\nu = 15$, $t = 8$, $\lambda^{-1} = 0.532632$, $\alpha = 0.53869147$, $\beta = 0.03107084$

par\time t	4	5	6	7	8	9	10
μ	6.2529	5.4956	5.2256	5.1247	5.1051	5.1313	5.1861
ν	7.5099	9.3873	11.2648	13.1423	15.0198	16.8972	18.7747
β_t	0.0281	0.0317	0.0324	0.0318	0.0306	0.0292	0.0277
100δ	9.43	2.18	4.22	2.33	$\boxed{1.39}$	6.00	10.97

Chapter 5

Conclusions

The introduction of sweeps consisting of alternating jumps and deletions, with time-invariant parameters ν and μ, provide us with an improved possibility of solving the fractionation model completely. We do announce such a solution, though it has much room for improvement. The exponential distribution of visible segment lengths could probably be established analytically, and it is also possible, though less likely, that the gamma distribution of invisible segment lengths could also be proved, including the α and β parameters as a function of the number of sweeps. Depending on the functional form of such a solution, the inference of t, μ and ν might be amenable through closed form formulae rather than the quadratic modeling. Nevertheless, we have succeeded for the first time in inferring the parameters of a fractionation model, albeit a "one-sided" model and a continuous analog of more realistic discrete fractionation models.

Aside from theoretical improvements, the first priority for this work should be the return to a discrete gene-order model of fractionation with the insights gained in the current report. This should be extended to, or at least tested on simulations of, two-sided fractionation models with subgenome dominance (higher deletion rates on

one copy of the genome than the other).

Appendix A

Calculation of C for Chapter 2.3

We calculate the changes in the C values counting the previous deletion points in a deletion interval $[x, x+a)$. $\Delta(i)$ represents the changing probability of the number of points with $C = i$. $\pi(i)$ represents that the number of deletions points which involve i previous deletion points. If the deletion point does not overlap any previous points, then $\Delta(1) = p_0$. If the deletion points overlap two points with probability p_2, and both points involve 1 previous point, then $\Delta(1) = -p_2\Big[2\big(\pi(1)\big)^2\Big]$. $\Delta(i)$ is the sum of all possible such events.

$$
\begin{aligned}
\Delta(1) &= p_0 - p_1\Big[\pi(1)\Big] \\
&\quad - p_2\Big[2\big(\pi(1)\big)^2 + 2\times\pi(1)\big(\sum_{i\neq 1}\pi(i)\big)\Big] \\
&\quad - p_3\Big[3\big(\pi(1)\big)^3 + 3\times\pi(1)\big(\sum_{i\neq 1}\pi(i)\big)^2 + 3\times 2\big(\pi(1)\big)^2\big(\sum_{i\neq 1}\pi(i)\big)\Big] \\
&\quad - p_4\Big[4\big(\pi(1)\big)^4 + 4\times\pi(1)\big(\sum_{i\neq 1}\pi(i)\big)^3 + 6\times 2\big(\pi(1)\big)^2\big(\sum_{i\neq 1}\pi(i)\big)^2 + 4\times 3\big(\pi(1)\big)^3\big(\sum_{i\neq 1}\pi(i)\big)\Big] \\
&\quad - p_5\Big[5\big(\pi(1)\big)^5 + 5\times\pi(1)\big(\sum_{i\neq 1}\pi(i)\big)^4 + 10\times 2\big(\pi(1)\big)^2\big(\sum_{i\neq 1}\pi(i)\big)^3 \\
&\qquad + 10\times 3\big(\pi(1)\big)^3\big(\sum_{i\neq 1}\pi(i)\big)^2 + 5\times 4\big(\pi(1)\big)^4\big(\sum_{i\neq 1}\pi(i)\big)\Big] - ... \\
&= p_0 - p_1\Big[\pi(1)\Big] \\
&\quad - p_2\Big[2\big(\pi(1)\big)^2 + 2\times\pi(1)\big(1-\pi(1)\big)\Big] \\
&\quad - p_3\Big[3\big(\pi(1)\big)^3 + 3\times\pi(1)\big(1-\pi(1)\big)^2 + 3\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)\Big] \\
&\quad - p_4\Big[4\big(\pi(1)\big)^4 + 4\times\pi(1)\big(1-\pi(1)\big)^3 + 6\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^2 + 4\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)\Big] \\
&\quad - p_5\Big[5\big(\pi(1)\big)^5 + 5\times\pi(1)\big(1-\pi(1)\big)^4 + 10\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^3 \\
&\qquad + 10\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)^2 + 5\times 4\big(\pi(1)\big)^4\big(1-\pi(1)\big)\Big] - ... \\
&\quad ... \\
&= p_0 - p_1\Big[\pi(1)\Big] \\
&\quad - p_2\Big[2\big(\pi(1)\big)^2 + 2\times\pi(1)\big(1-\pi(1)\big)\Big] \\
&\quad - p_3\Big[3\big(\pi(1)\big)^3 + 3\times\pi(1)\big(1-\pi(1)\big)^2 + 3\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)\Big] \\
&\quad - p_4\Big[4\big(\pi(1)\big)^4 + 4\times\pi(1)\big(1-\pi(1)\big)^3 + 6\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^2 + 4\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)\Big] \\
&\quad - p_5\Big[5\big(\pi(1)\big)^5 + 5\times\pi(1)\big(1-\pi(1)\big)^4 + 10\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^3 \\
&\qquad + 10\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)^2 + 5\times 4\big(\pi(1)\big)^4\big(1-\pi(1)\big)\Big] - ... \\
&\quad ... \\
&= p_0 - p_1\Big[\pi(1)\Big] - 2p_2\Big[\pi(1)\Big] - 3p_3\Big[\pi(1)\Big] - 4p_4\Big[\pi(1)\Big] - ...
\end{aligned}
\tag{A.1}
$$

where, renaming the multiplier of p_2 as $\mathbf{P}_2$, that of p_3 as $\mathbf{P}_3$,...,

$$
\begin{aligned}
\mathbf{P}_2 &= 2\big(\pi(1)\big)^2 + 2\times\pi(1)\big(1-\pi(1)\big)\\
&= 2\Big[\big(\pi(1)\big)^2 + \pi(1) - \big(\pi(1)\big)^2\Big]\\
&= 2\pi(1)\\
\mathbf{P}_3 &= 3\big(\pi(1)\big)^3 + 3\times\pi(1)\big(1-\pi(1)\big)^2 + 3\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)\\
&= 3\Big[\big(\pi(1)\big)^3 + \big(1-\pi(1)\big)\Big[\pi(1) - \big(\pi(1)\big)^2 + 2\big(\pi(1)\big)^2\Big]\Big]\\
&= 3\Big[\big(\pi(1)\big)^3 + \pi(1)\Big[1 - \big(\pi(1)\big)^2\Big]\Big]\\
&= 3\pi(1)\\
\mathbf{P}_4 &= 4\big(\pi(1)\big)^4 + 4\times\pi(1)\big(1-\pi(1)\big)^3 + 6\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^2 + 4\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)\\
&= 4\Big[\big(\pi(1)\big)^4 + \pi(1)\big(1-\pi(1)\big)^2\Big[1-\pi(1)+3\pi(1)\Big] + 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)\Big]\\
&= 4\Big[\big(\pi(1)\big)^4 + \pi(1)\big(1-\pi(1)\big)\Big[1+\pi(1) - 2\big(\pi(1)\big)^2 + 3\big(\pi(1)\big)^2\Big]\Big]\\
&= 4\Big[\pi(1)\Big[1+\pi(1) + \big(\pi(1)\big)^2 - \pi(1) - \big(\pi(1)\big)^2 - \big(\pi(1)\big)^3 + \big(\pi(1)\big)^3\Big]\Big]\\
&= 4\pi(1)\\
\mathbf{P}_5 &= 5\big(\pi(1)\big)^5 + 5\times\pi(1)\big(1-\pi(1)\big)^4 + 10\times 2\big(\pi(1)\big)^2\big(1-\pi(1)\big)^3 + 10\times 3\big(\pi(1)\big)^3\big(1-\pi(1)\big)^2\\
&\quad + 5\times 4\big(\pi(1)\big)^4\big(1-\pi(1)\big)\\
&= 5\Big[\big(\pi(1)\big)^5 + \pi(1)\big(1-\pi(1)\big)^3\Big[1-\pi(1)+4\pi(1)\Big] + 6\big(\pi(1)\big)^3\big(1-\pi(1)\big)^2 + 4\big(\pi(1)\big)^4\big(1-\pi(1)\big)\Big]\\
&= 5\Big[\big(\pi(1)\big)^5 + \pi(1)\big(1-\pi(1)\big)^2\Big[1+2\pi(1) - 3\big(\pi(1)\big)^2 + 6\big(\pi(1)\big)^2\Big] + 4\big(\pi(1)\big)^4\big(1-\pi(1)\big)\Big]\\
&= 5\Big[\big(\pi(1)\big)^5 + \pi(1)\big(1-\pi(1)\big)\Big[1+2\pi(1) + 3\big(\pi(1)\big)^2 - \pi(1) - 2\big(\pi(1)\big)^2 - 3\big(\pi(1)\big)^3 + 4\big(\pi(1)\big)^3\Big]\Big]\\
&= 5\Big[\pi(1)\Big[1+\pi(1) + \big(\pi(1)\big)^2 + \big(\pi(1)\big)^3 - \pi(1) - \big(\pi(1)\big)^2 - \big(\pi(1)\big)^3 - \big(\pi(1)\big)^4 + \big(\pi(1)\big)^4\Big]\Big]\\
&= 5\pi(1)
\end{aligned}
\tag{A.2}
$$

Similarly,

$$\Delta(2) = p_1\Big[\pi(1)\Big] - p_1\Big[\pi(2)\Big] - 2p_2\Big[\pi(2)\Big] - 3p_3\Big[\pi(2)\Big] - 4p_4\Big[\pi(2)\Big] - ... \tag{A.3}$$

$$\Delta(3) = p_1\Big[\pi(2)\Big] + p_2\Big[\big(\pi(1)\big)^2\Big] - p_1\Big[\pi(3)\Big] - 2p_2\Big[\pi(3)\Big] - 3p_3\Big[\pi(3)\Big] - 4p_4\Big[\pi(3)\Big] - ... \tag{A.4}$$

$$\Delta(4) = p_1\Big[\pi(3)\Big] + 2p_2\Big[\pi(1)\pi(2)\Big] + p_3\Big[\big(\pi(1)\big)^3\Big] \tag{A.5}$$
$$- p_1\Big[\pi(4)\Big] - 2p_2\Big[\pi(4)\Big] - 3p_3\Big[\pi(4)\Big] - 4p_4\Big[\pi(4)\Big] - ...$$

$$\Delta(5) = p_1\Big[\pi(4)\Big] + p_2\Big[2\pi(1)\pi(3) + \big(\pi(2)\big)^2\Big] + 3p_3\Big[\big(\pi(1)\big)^2\pi(2)\Big] + p_4\Big[\big(\pi(1)\big)^4\Big] \tag{A.6}$$

$$- p_1\Big[\pi(5)\Big] - 2p_2\Big[\pi(5)\Big] - 3p_3\Big[\pi(5)\Big] - 4p_4\Big[\pi(5)\Big] - 5p_5\Big[\pi(5)\Big] - ... \tag{A.7}$$

...

Therefore,

$$\begin{aligned}
\sum_{i=1}^{\infty}\Delta(i) = p_0 \; &+ p_2\Big[-2\big(\pi(1)+\pi(2)+\pi(3)+...\big) + \big(\pi(1)\big)^2 + 2\pi(1)\pi(2) + \big(\pi(2)\big)^2 \\
&\quad + 2\pi(1)\pi(3) + ...\Big] \\
&+ p_3\Big[-3\big(\pi(1)+\pi(2)+\pi(3)+...\big) + \big(\pi(1)\big)^3 + 3\big(\pi(1)\big)^2\pi(2) + 3\big(\pi(1)\big)^2\pi(3) \\
&\quad + 3\pi(1)\big(\pi(2)\big)^2 + ...\Big] \\
&+ p_4\Big[-4\big(\pi(1)+\pi(2)+\pi(3)+...\big) + \big(\pi(1)\big)^4 + ...\Big] + ... \\
&+ ... \\
&= p_0 + p_2\Big[-2 + \big(\pi(1)+\pi(2)+\pi(3)+...\big)^2\Big] \\
&\quad + p_3\Big[-3 + \big(\pi(1)+\pi(2)+\pi(3)+...\big)^3\Big] \\
&\quad + p_4\Big[-4 + \big(\pi(1)+\pi(2)+\pi(3)+...\big)^4\Big] + ... \\
&\quad + ... \\
&= p_0 - p_2 - 2p_3 - 3p_4 - ...
\end{aligned} \tag{A.8}$$

Appendix B

Simulation algorithms for Chapter 3

B.1 Discrete Fractionations Model

In this discrete fractionations model, the genome sequence is represented by a set $S_i = \{s_1, s_2, \ldots, s_l\}$, where s_i is an integer same to the number of iteration where it is deleted. For example, if the s_i is deleted in the 1^{st} iteration, then $s_i = 1$. If it is undeleted after all iteration, its final value will be 0 since $S_0 = \{0, 0, \ldots, 0\}$ is the genome sequence in 0^{th} iteration which can also be writen as the original sequence.

Algorithm 1 will set up the original sequence with given length. The sequence will run the geometric fractionation processes, which is defined in Algorithm 2, by numbers of iterations, which we can defined to be t. The final sequence will be sent back to Algorithm 1. The final sequence will be a sequence consists of intergers $\{0, 1, \ldots, t\}$. Then it will be calculated the length of undeleted fraction of a discrete genome sequence after t iterations, by calculate the length of serial "0"s in the

sequence. We say the undeleted sequence to be "0" or "visible",which is what we concerned, and the deleted sequence to be "$\neq 0$" or "invisible".

Algorithm 1: DISCRETE finds the length of visible sequence after fractionation.

Input: Two integers *iteration* and L, which represent the number of iterations and length of undeleted genome sequence. Two denominations μ and ν, which is the means of X and A.

Output: A integer λ, which is the lengths of visible sequence after fractionations.

```
S ← new Array with length = L; Set up a brand new genome sequence
for i ← 1 to L do
    s_i ← 0
for i ← 1 to iteration do
    S ← S_new, which call Algorithem 2
    l ← lengthofS_new
λ ← 0
for i ← 1 to l do
    if s_i = 0 then
        λ = λ + 1;
    else
        if λ ≠ 0 then
            return λ
            λ = 0;
```

Algorithm 2 defines one iteration following a fractionation model with several delation steps. It will simulate a new genome sequence by fractioning the remaining from previous iteration. Before each iteration, to keep the total length of visible sequence same to the length of original sequence and to keep the sequence's "memorization", the remaing will be extended by adding the copy of its beginning with both invisible and visible sequences. In this j^{th} iteration, the set X_i and A_i is generated from Algorithm 3.

In Algorithm 2, the sequence will be deleted from a deletion point x_1, and be

deleted by length a_1. Then the deletion process walk along the sequence to the second deletion point x_2 and start a new deletion a_2. Therefore, the location of i^{th} deletion is $x_{i-1} + a_{i-1} + x_i$ in this memoriable sequence. The process keeps going until out of the whole sequence. The deletion points and lengths of j^{th} iteration are represents by integer sets $X_i = \{x_1, x_2, \ldots, x_n\}$ and $A_i = \{a_1, a_2, \ldots, a_m\}$, where X and A are from Geometric Distribution with given mean ν and μ, respectively.

Algorithm 3 simulates one random number from Geometric Distribution with given mean μ based on the Cumulative Distribution Function (CDF):

$$F(x) = 1 - (1-p)^x, p = \frac{1}{mean} \tag{B.1}$$

B.2 Continuous Fractionations Model

In the continuous fractionations model, the genome sequence is represented by a set $S_i = \{(s_1, s_2), (s_3, s_4), \ldots, (s_{2t-1}, s_{2t}\}$, where t is an integer greater than 0. For any integer $i > j > 0$, then $s_i > s_j \geq 0$; and (s_{2i-1}, s_{2i}) represents one remaining genome fraction with length $s_{2i} - s_{2i-1}$; t is the number of genome fractions. The gap between the fractions is the deleted fractions.

Algorithm 4 will first generate an new continuous genome sequence with given length. Then this genome sequence will sent to Algorithm 5 by t iterations, where the genome sequence is deleted in the exponential fractionation processes. Then the Algorithm 4 will calculate the length of all remaining undeleted fractions by calculating $s_{2i-1} - s_{2i}$, where i is an integer and $0 \leq i \leq t$.

Algorithm 5 defines one j^{th} iteration following a exponential fractionation model

Algorithm 2: Simulation of Discrete Sequence at j^{th} iteration

Input: A set of integers $S_{j-1} = \{s_1, s_2, \ldots, s_l\}$ represents the visible and invisable genome sequence from one step previous $(j-1)$ iteration. A constant integer L represent the length of original undeleted sequence. An integer l represent the total length of remaining. Two denominations ν and μ are used to simulated X and A.

Output: An updated genome sequence after this j^{th} iteration.

$x \sim Geometric(\nu); a \sim Geometric(\mu)$; Call Algorithm 3
$index \leftarrow 0$; $x_1 = 0$
for $i \leftarrow 1$ ***to*** $length$ **do**
    if $s_i = 0$ **then**
        $index = index + 1$; To account the length of undeleted sequence in this remaining.;
$left \leftarrow L - index$
$num \leftarrow 0; i \leftarrow 1$
while $num < left$ **do**
    if $s_i = 0$ **then**
        $num = num + 1$
    $i = i + 1$;
$l_{new} = l + num$
$S_{new} = \{sn_1, sn_2, \ldots, sn_{ln}\}$ with dimention l_{new}; Set up a new sequence by adding up the length of undeleted sequence to L
$index \leftarrow 1$
for $i \leftarrow 1$ ***to*** l_{new} **do**
    if $i < l$ **then**
        $sn_i \leftarrow s_i$;
    else
        $sn_i \leftarrow s_{index}$
        $index = index + 1$
while $x + a < L$ **do**
    $index = 1$
    for $i \leftarrow 1$ ***to*** l_{new} **do**
        if $sn_i = 0$ **then**
            $index = index + 1$
            if $index \geq x \wedge index < x + a$ **then**
                $sn_i = j$;
    $x_1 \sim Geometric(\nu)$; $x \leftarrow x + a + x1$
    $a \sim Geometric(\mu)$;
return S_{new}

Algorithm 3: GEOMETRIC DISTRIBUTION Simulate a random number from geometric distribution.

Input: A denomination μ is the mean of geometric distribution.
Output: A random integer simulated from geometric distribution.

$p \leftarrow 1/mean$
$rand \sim N(0,1)$
$n \leftarrow \frac{\log(1-rand)}{\log(1-p)}$
$x \approx n$, where x is an integer
return x

Algorithm 4: CONTINUOUS finds the length of undeleted sequence after fractionations.

Input: A integer *interation* represents the numbers of iterations. Three positive real number L, μ, ν, represents the defined totoal lengths of undeletions, mean of A and X, respectively.
Output: The length of visible undeleted sequence after exponential fractions.

$S \leftarrow$ Array $\{0, L)\}$; which is defined as the original genome sequence
$T \leftarrow 2$; which is the length of Array S
for $j \leftarrow 1$ ***to*** *iteration* **do**
    $S \leftarrow S_{new}$, call Algorithm 5
    $T \leftarrow$ the length of the updated genome sequence S;
for $i \leftarrow 1$ ***to*** $T/2$ **do**
    $\lambda \leftarrow s_{2i} - s_{2i-1}$
    return λ

with multiple delation steps. Before starting the deletion, the length of undeleted fractions of sequence will be extended to constant L, by adding copy of partial sequence from the begining to the end of the sequence, which is concerned in Algorithm 6. Then each deletion i will start from a deletion point, called x_i, and a deletion length, called a_i, which are generated from Algorithm 7. The real numbers $x_i \in X \sim Exp(\nu)$ and $a_i \in A \sim Exp(\mu)$. Since the total length of undeleted sequence is L, we have

$$\sum_{i=1}^{n} x_i + a_i \leq L \tag{B.1}$$

where the n is the number of deletions in j^{th} iteration.

For every fraction (s_{2i-1}, s_{2i}), we define that s_{2i-1} to be "start point" and s_{2i} to be "end point" of the fraction. When the deletion starts, the algorithm will add up the length of fractions from begining, which we defined to be variable k, until $k \geq x_1$. We say this stops at fraction (s_{2i-1}, s_{2i}), then location of deletion point on this memorable sequence is $s_{2i} - (k - x_1)$, and it becomes the new "end point" of this fraction as s_{2i}. Then we defined a variable m to count the length of deletion and it is initialized as $k - x_1$, which is the leftover of the current fraction. As long as $m < a_1$ the length of deletion, it should be added up by the length of following fraction. We say this stops at fraction (s_{2j-1}, s_{2j}), where m just greater or equal to a_1, and $j \geq i$. Then the location of the end of deletion is $s_{2j} - (m - a_1)$, and it becomes the new "start point" of this fraction as $s_{2j'-1}$, where $j' = i + 1$. The second deletion will start with a new x_2, a_2 simulated from Algorithm 7. It follows similar steps as the first deletion except that k is initialized to $s_{2j'-1}$ and is added up from this fraction $(s_{2j'-1}, s_{2j'})$. The fractionation stops when it run out of the bound.

Algorithm 7 simulate a random number from Exponential Distribution with given

Algorithm 5: SIMULATION OF CONTINUOUS SEQUENCE at j^{th} iteration.

Input: A set of intervels $S_{j-1} = \{(s_1, s_2), (s_3, s_4), \ldots, (s_{2l-1}, s_{2l})\}$ represents the visible genome sequence from previous $(j-1)$ iteration with length $2l$. A real number L is the constant length of undeleted sequence. Two denominations ν and μ are used to simulate X and A.

Output: The updated genome sequence S_j after this j^{th} iteration.

$startN = endN = 0$; $startP = endP = 1$
$x \sim Geometric(\nu)$; $a \sim Geometric(\mu)$; Call Algorithm 7
$S_{new} = \{(sn_1, sn_2), \ldots, (sn_{2ln-1}, sn_{2ln})\} \leftarrow$ New Array with length $2l_{new} = 2l + 2i - 2$; Call Algorithm 6
$k = 0$;
while $x + a < L$ **do**
    $i \leftarrow \frac{startP+1}{2}$;
    $k \leftarrow endN$; $x_p \leftarrow k + x_p$;
    while $k < x_p$ **do**
        $k \leftarrow k + sn_{2i} - sn_{2i-1}$;
        $i = i + 1$;
    $i = i - 1$; $startP \leftarrow 2i$;
    $startN \leftarrow sn_{2i} - k + x_p$;
    $m \leftarrow sn_{2i} - startN$; $j \leftarrow i$;
    while $m < a$ **do**
        $j = j + 1$;
        $m \leftarrow m + sn_{2j} - sn2j - 1$;
    $endP \leftarrow 2j - 1$;
    $endN \leftarrow sn_{2j} - m + a$;
    $S_{deleted} = \{(sd_1, sd_2), (sd_3, sd_4), \ldots, (sd_{2ld-1}, sd_{2ld})\} \leftarrow$ New Array with length $2l_{deleted}$;
    $2l_{deleted} = 2l_{new} - endP + startP + 1$;
    for $i \leftarrow 1$ ***to*** $startP - 1$ **do**
        $sd_i \leftarrow sn_i$;
    $sd_{startP} \leftarrow startN$; $sd_{startP+1} \leftarrow endN$;
    $j = 0$;
    for $i \leftarrow startP + 2$ ***to*** $2ld$ **do**
        $sd_i \leftarrow snendP + 1 + j$;
        $j = j + 1$;
    $S_{new} \leftarrow S_{deleted}$; $l_{new} \leftarrow l_{deleted}$;
    $x_p \leftarrow$ Call Algorithm 7 with mean ν; $x \leftarrow x + a + x_p$;
    $a \leftarrow$ Call Algorithm 7 with mean μ;
return S_{new}

Algorithm 6: ADDING UP the length of visible sequence to L.

Input: An integer L is the total length of visible sequence. A set $S = \{(s_1, s_2), (s_3, s_4), \ldots, (s_{2l-1}, s_{2l})\}$ represent the genome sequence from $j-1$ iteration

Output: A new memeorable sequence S_{new} with L visiable sequence.

$account = 0$; This is used to account the length of S_{j-1}
for $t \leftarrow 1$ ***to*** $2l$ **do**
    $account = account + (s_{2t} - s_{2t-1})$;
$index = L - account$
$num = 0$; $i = 0$;
while $num \leq index$ **do**
    $i = i + 1$;
    $num = num + s_{2i} - s_{2i-1}$;
$S_{new} = \{(sn_1, sn_2), \ldots, (sn_{2ln-1}, sn_{2ln})\} \leftarrow$ New Array with length $2l_{new} = 2l + 2i - 2$;
$j = 2$; **for** $i \leftarrow 1$ ***to*** $2l - 1$ **do**
    $sn_i \leftarrow s_i$;
$sn_{2l} = s2l + (s_j - s_1)$;
$j = j + 1$; **for** $i \leftarrow 2l + 1$ ***to*** $2ln$ **do**
    $sn_i \leftarrow s_{2l} + s_j$;
    $j = j + 1$;
$sn_{2ln} \leftarrow sn_{2ln} - (num - index)$;
return S_{new}

mean μ by the CDF function:

$$F(x) = 1 - \exp^{-\frac{x}{\mu}} \tag{B.2}$$

, which $F(x)$ is random number from Normal Distribution $(0, 1)$.

Algorithm 7: EXPONENTIAL DISTRIBUTION simulate a random number from Exponential Distribution with given mean μ.

Input: One real number μ
Output: A random number $x \sim Exp(\mu)$
1 $rand \sim N(0, 1)$;
2 $x \leftarrow -\mu * \ln(1 - rand)$;
3 **return** x

Bibliography

[1] Sankoff, D., Zheng, C. and Zhu, Q. (2010) The collapse of gene complement following whole genome duplication. *BMC Genomics* **11,** 313.

[2] Wang, B., Zheng, C. and Sankoff, D. (2011) Fractionation statistics. *BMC Bioinformatics* **12,** S9:S5.

[3] Sankoff, D., Zheng, C. and Wang, B. (2012) A model for biased fractionation after whole genome duplication. *BMC Genomics* **13,** S1:S8.

[4] Sankoff, D., Zheng, C., Wang,B. and Buen Abad Najar, C.F. (2014) Structural vs. functional mechanisms of duplicate gene loss following whole genome doubling. *manuscript*

[5] Byrnes, J.K., Morris, G.P. and Li, W.H. (2006) Reorganization of adjacent gene relationships in yeast genomes by whole-genome duplication and gene deletion. *Molecular Biology and Evolution* **23,** 1136–1143.

[6] van Hoek, M.J. and Hogeweg, P. (2007) The role of mutational dynamics in genome shrinkage. *Molecular Biology and Evolution* **24,** 2485–2494.

[7] Zheng, C., Wall, P.K., Leebens-Mack, J., dePamphilis, C., Albert, V.A. and Sankoff, D. (2009) Gene loss under neighbourhood selection following whole genome duplication and the reconstruction of the ancestral *Populus* diploid. *Journal of Bioinformatics and Computational Biology* **7,** 499-520.