

How Long it Takes to Complete the Collection Log

Platpus3000

August 15, 2025

1 Drop Rates

Every RuneScape player is familiar with the geometric distribution, whether they know it or not. We have all killed some boss or slayer monster hundreds or thousands of times (or more) chasing that special drop: an abyssal whip, a berserker ring, a tanzanite fang. In each instance¹ the number of kills you do before receiving your desired drop is *geometrically distributed*; that is, the probability of doing a given number of kills without receiving the drop decays geometrically². In common RuneScape parlance, one would say that the probability of *going dry* (not getting a desired drop) decays geometrically in the number of kills.

For example, the abyssal whip is dropped by abyssal demons at a rate of $1/512$. This means that every time you kill an abyssal demon, you receive a whip with probability $p = 1/512$, independent of other kills. What is the probability of going n kills dry? that is, what is the probability of doing n kills and not receiving a whip on any of them? Let D_n denote this event; we want to find $P(D_n)$.

First, we note that if $n = 1$, then we are asking what is probability of not getting a whip on the first kill. This event is the complement of getting a whip on the first kill, so it is given by $P(D_1) = 1 - p = 1 - 1/512 = 511/512 = q$.

Second, we note that you will go dry $n + 1$ kills if and only if you go dry n kills and then fail to get a whip on the $(n + 1)$ th kill. Thus, if A_{n+1} is the event that you receive a whip on the $(n + 1)$ th kill, we have $D_{n+1} = D_n \cap A_{n+1}^C$. Here, A_{n+1}^C is the complement of A_{n+1} , so $P(A_{n+1}^C) = 1 - P(A_{n+1}) = 1 - p = q$ because $P(A_{n+1}) = p$ is the probability of getting a whip on a particular kill. Since whether you get a whip on kill $n + 1$ or not is independent of whether you got one in the first n kills or not, it follows that D_n and A_{n+1} are independent. Then for any $n \geq 1$,

$$P(D_{n+1}) = P(D_n \cap A_{n+1}) = P(D_n)P(A_{n+1}) = qP(D_n). \quad (1)$$

¹It is geometrically distributed in most instances, anyway; for example, the number of kills to obtain ring upgrade items dropped by Desert Treasure II bosses is not geometrically distributed.

²Equivalently, one can say that the probability decays *exponentially*.

It is easy to show that (1) implies that

$$P(D_n) = q^n. \quad (2)$$

For example, consider $P(D_5)$. Applying (1) repeatedly gives $P(D_5) = qP(D_4) = q^2P(D_3) = q^3P(D_2) = q^4P(D_1) = q^5$. A proof by induction shows that this pattern holds for any n . Since $0 < q < 1$, it follows that $P(D_n) \rightarrow 0$ geometrically as $n \rightarrow \infty$, as we claimed.

Suppose you get your first whip drop on the N th kill; thus, N is a *discrete random variable*, and we can ask “what is the distribution of N ”? The answer to this question is easy. For $n \geq 1$, we need to compute $P(N=n)$. If $n = 1$, then $P(N=1) = P(A_1) = p$ is the probability of getting an abyssal whip on the first kill. If $n > 1$, then $P(N=n)$ is the probability of going dry for $n - 1$ kills and then getting a whip on the n th kill; that is, $\{N=n\} = D_{n-1} \cap A_n$, so

$$P(N=n) = P(D_{n-1} \cap A_n) = P(D_{n-1})P(A_n) = pq^{n-1}. \quad (3)$$

Conveniently, this formula also works when $n = 1$, so $P(N=n) = pq^{n-1}$ for $n \geq 1$.

A random variable with this distribution is called a *geometric random variable*, or said to be geometrically distributed with parameter p . In general, a geometric random variable describes the number of attempts needed to achieve some desired outcome when repeatedly attempting something in which the desired outcome occurs independently and with equal probability p on each attempt. For this reason, the number of kills needed to get a desired drop is almost always a geometric random variable.

Since (3) provides a simple and explicit formula for the distribution of N , it is easy to answer virtually any question that can be asked about N . For example, what is the expected value, or average, of N ? This is given by

$$\mathbb{E}[N] = \sum_{n=1}^{\infty} nP(N=n) = \sum_{n=1}^{\infty} npq^{n-1} = p \sum_{n=1}^{\infty} \frac{d}{dx} [x^n]_{x=q} = p \frac{d}{dx} \left[\frac{x}{1-x} \right]_{x=q} \quad (4)$$

because the series $\sum_{n=1}^{\infty} nx^{n-1}$ converges uniformly and absolutely in a neighborhood of q (by the ratio test, among others), and the geometric series $\sum_{n=1}^{\infty} q^n$ converges. Thus,

$$\mathbb{E}[N] = p \frac{(1-q) + q}{(1-q)^2} = \frac{p}{p^2} = \frac{1}{p}. \quad (5)$$

In the case of the abyssal whip, we have that the average kill on which one first receives a whip is $\mathbb{E}[N] = \frac{1}{1/512} = 512$, as one would expect. More generally, if the probability of getting a drop is $p = 1/m$, then $\mathbb{E}[N] = m$.

We can ask some other interesting questions, as well, which do not have such obvious answers. For example, how likely are you to receive a whip in 512 kills? Certainly it isn't guaranteed, even though 512 is the average number of kills needed to receive a whip. Translating the question, we can equivalently ask what is the probability that $N \leq 512$? Since this event is the disjoint union of the event that $N = 1$ or $N = 2$ or...or $N = 512$, we can write

$$\begin{aligned} P(N \leq 512) &= P\left(\bigcup_{n=1}^{512} N=n\right) = \sum_{n=1}^{512} P(N=n) \\ &= \sum_{n=1}^{512} pq^{n-1} = p \frac{1-q^{512}}{1-q} = 1 - (1-1/512)^{512} \\ &\approx 63.2\%. \end{aligned} \tag{6}$$

To the astute calculus student, the last expression should look familiar—it is very close to $(1 + 1/k)^k$, with $k = 512$, which is the sequence from which the number e is derived in the limit as $k \rightarrow \infty$. Indeed, if the probability p of getting a drop is pretty small, then $\frac{1}{p}$ is pretty large, and, repeating the same calculation as above, we have

$$P\left(N \leq \frac{1}{p}\right) = \sum_{n=1}^{\lfloor \frac{1}{p} \rfloor} pq^{n-1} = p \frac{1-q^{\lfloor \frac{1}{p} \rfloor}}{1-q} = 1 - (1-p)^{\lfloor \frac{1}{p} \rfloor}.$$

Letting $k = \frac{1}{p}$, we get

$$P\left(N \leq \frac{1}{p}\right) = 1 - (1-1/k)^{k+(\lfloor k \rfloor - k)}.$$

Since we assumed that p is pretty, small we can now estimate $P\left(N \leq \frac{1}{p}\right)$ by the limit as $p \rightarrow 0$. If $p \rightarrow 0$, then $k \rightarrow \infty$; it is clear that $|\lfloor k \rfloor - k| \leq 1$, so $(1-1/k)^{\lfloor k \rfloor - k} \rightarrow 1$ as $p \rightarrow 0$ and $k \rightarrow \infty$. Thus,

$$\lim_{p \rightarrow 0} P\left(N \leq \frac{1}{p}\right) = 1 - \lim_{k \rightarrow \infty} (1-1/k)^k = 1 - e^{-1} \approx 63.2\% \tag{7}$$

This means that, as the drop rate of an item gets very small, the chance of getting the item on rate approaches $1 - e^{-1}$ exactly. Moral of the story: no matter what you are doing, it is actually quite likely (around 36.8% chance) that you will go dry past the drop rate, so don't be surprised when it happens!

As is often the case with probability, seemingly straightforward questions may have surprising answers; we just saw an example. Another question with a surprising answer: what is the *median* number of kills to obtain an abyssal whip (or whatever drop you are going for)? By the median we mean the number of kills m such that

half of the time you need to do fewer than m kills and half the time you need to do more than m kills to get a whip. Intuition suggests that it should be close to the average number of kills, but we already know that you need to do fewer than the average number of kills to get a whip around 63.2% of the time, so the median must be less than the average.

Mathematically, the definition of the median implies that $P(N \leq m) = \frac{1}{2}$. We know that $P(N \leq m) = (1 - 1/512)^m$ (substituting m for 512 above); solving for m gives

$$m = \frac{\log(1/2)}{\log(1 - 1/512)} \approx 355, \quad (8)$$

so half of the time you will get a whip in 355 kills or fewer, and the other half of the time you will get one in more than 355 kills.

The formula above might not be an integer, but we can round it down to get the largest value of m such that $P(N \leq m) \leq \frac{1}{2}$, and we can round it up to get the smallest value of m such that $P(N \leq m) \geq \frac{1}{2}$. In general, we see that

$$m = \frac{\log(1/2)}{\log(1 - p)}. \quad (9)$$

The astute calculus student should also know that if the probability p of getting a drop is small, then we can estimate $\log(1 - p)$ using the Taylor series centered at 1, which gives:

$$\log(1 - p) = -p - \frac{p^2}{2} - \frac{p^3}{3} + \mathcal{O}(p^4).$$

Then

$$m = \frac{\log(1/2)}{-p - \frac{p^2}{2} + \mathcal{O}(p^3)} = \frac{\frac{1}{p} \cdot \log(2)}{1 + \frac{p}{2} + \mathcal{O}(p^2)},$$

which is the sum of the geometric series $\frac{1}{p} \cdot \log(2) \sum_{k=0}^{\infty} (-\frac{p}{2} + \mathcal{O}(p^2))^k$. Thus,

$$m = \frac{1}{p} \cdot \log(2) \left(1 - \frac{p}{2} + \mathcal{O}(p^2)\right) \approx (1/p - 1/2) \log(2) = (\mathbb{E}[N] - 1/2) \log(2)$$

when p is pretty small. Substituting $\mathbb{E}[N] = 512$ for the whip once again gives $m \approx 355$. Moreover, if we look at the ratio of the median to the mean, we see that

$$\frac{m}{\mathbb{E}[N]} = \frac{\frac{1}{p} \cdot \log(2) \left(1 - \frac{p}{2} + \mathcal{O}(p^2)\right)}{\frac{1}{p}} \rightarrow \log(2) \approx 0.693 \quad (10)$$

as $p \rightarrow 0$. Thus, for pretty much any drop rate, there is a 50% chance of getting the drop in about 69.3% of the expected number of kills, which is a somewhat unexpected result. While the previous calculation showed that going dry past the

drop rate is not that uncommon, this result, in contrast, shows that it is also very common to get a drop well under the drop rate.

This fact can be accounted for nicely by understanding how much variability there is in the number of kills needed to get a desired drop. Mathematically, this is best understood through the variance and standard deviation of the variable N . Generally speaking, a random variable will take values within plus or minus a few standard deviations of the average with a pretty high probability. What is the standard deviation of N ? We can perform this calculation fairly easily as well. First we compute the variance of N :

$$\begin{aligned}
\text{Var}[N] &= \mathbb{E}[N^2] - (\mathbb{E}[N])^2 \\
&= -\frac{1}{p^2} + \sum_{n=1}^{\infty} n^2 P(N=n) \\
&= -\frac{1}{p^2} + \sum_{n=1}^{\infty} n^2 pq^{n-1} \\
&= -\frac{1}{p^2} + p \sum_{n=1}^{\infty} \frac{d^2}{dx^2} [x^{n+1}]_{x=q} - \frac{d}{dx} [x^n]_{x=q} \\
&= -\frac{1}{p^2} + p \frac{d^2}{dx^2} \left[\sum_{n=1}^{\infty} x^{n+1} \right]_{x=q} - p \frac{d}{dx} \left[\sum_{n=1}^{\infty} x^n \right]_{x=q} \\
&= -\frac{1}{p^2} + p \frac{d^2}{dx^2} \left[\frac{x^2}{1-x} \right]_{x=q} - p \frac{d}{dx} \left[\frac{x}{1-x} \right]_{x=q} \\
&= -\frac{1}{p^2} + p \frac{d}{dx} \left[\frac{2x - x^2}{(1-x)^2} \right]_{x=q} - p \frac{(1-q) + q}{(1-q)^2} \\
&= -\frac{1}{p^2} - \frac{1}{p} + p \frac{2(1-q)(1-q)^2 + 2(2q - q^2)(1-q)}{(1-q)^4} \\
&= -\frac{1}{p^2} - \frac{1}{p} + \frac{2 - 4q + 2q^2 + 4q - 2q^2}{p^2} \\
&= \frac{1-p}{p^2} = \frac{q}{p^2}.
\end{aligned} \tag{11}$$

Note that the exchange of summation and differentiation is again valid due to the uniform convergence in a neighborhood of q of all series involved. Then the standard deviation of N is $\sigma = \sqrt{\text{Var}[N]} = \frac{\sqrt{q}}{p}$. If p is pretty small, then, by the binomial theorem, we have

$$\sigma = \frac{\sqrt{1-p}}{p} = \frac{1 - \frac{p}{2} + \mathcal{O}(p^2)}{p} = \frac{1}{p} - \frac{1}{2} + \mathcal{O}(p) = \mathbb{E}[N] - \frac{1}{2} + \mathcal{O}(p), \tag{12}$$

so, again, if the drop rate p is pretty small, as it usually is, then the standard deviation of N is nearly the same as the average (minus $1/2$). Indeed, for the abyssal whip, the standard deviation is $\sigma \approx 511.4998$. This suggests that the number of kills to get an item can vary widely; the fact that it is only possible to go one standard deviation less than the average suggests that a considerable amount of probability is concentrated greater or much greater than the average, meaning that it is not too uncommon to go dry very far above the drop rate (2, 3 or 4 times, even), but also, on the other hand, it is quite common to get a drop well before reaching the average number of kills.

Actually, one could say that the fact that $\sigma \approx \mathbb{E}[N]$ means that the average $\mathbb{E}[N]$ is almost meaningless, as the amount of uncertainty in that estimate is roughly 100%. The takeaway here is that drop rates are *very* rough estimates of the number of kills needed to get a drop, and you should almost always expect the actual number of kills to be much different.

2 Collection Logging

The release of the *collection log*, which tracks the number of virtually every significant item that a player has ever received, has stimulated much interest in *collection logging*, which means attempting to get at least one of every item tracked by the collection log. In reality—as we will see—this is nearly impossible, but because the log is divided into sections that turn green when the player has received at least one of every item in the section, players are still interested in “green-logging” as many sections as possible. This is a much more reasonable goal, as many sections can be completed in a manageable (if still large) amount of time.

Just how much time does it take to complete a section of the collection log? We show how to answer this question here, attempting to provide as much detail and nuance as we did in section 1 for single-item drop rates.

Each section of the collection log contains items that are obtained from the same source, which we will model as follows. Suppose that n distinct items are given, which we will refer to by the numerical labels $1, 2, \dots, n$, and suppose that X_k is the item obtained on drop k , which has distribution

$$P(X_k=i) = p_i, \quad i = 0, 1, 2, \dots, n.$$

Note that $X_k = 0$ indicates that none of the n items were obtained on drop k —in other words, 0 is the label for “junk” drops. We assume that $\{X_k\}_{k=1}^\infty$ are independent and identically distributed³.

³This model covers the vast majority of collection log sections. As we mentioned at the start, there are, however, a few cases of newer activities in which this simple model is not correct. We will have to handle such cases separately.

Now let us suppose that you wish to obtain t_1 drops of item 1, t_2 drops of item 2, and so on. Denote by $t = (t_1, t_2, \dots, t_n)$ the *target vector*, which consists of your target quantities for each item. Thus, the target vector for green-logging, or getting at least one of each item, is $t = (1, 1, \dots, 1)$. We now pose a similar question to what we asked in section 1: how many kills are necessary to obtain the target number of items? We have posed this question in a bit more generality (as opposed to just asking about green-logging) because there are situations in which it is desirable to get more than one of some items⁴. For example, irons generally want to obtain four ancient crystals from revenants in order to build wilderness obelisks in their player-owned houses.

Assuming that $|t| \geq 1$ (we want to obtain at least one drop), we are interested in the variable

$$M(t) = \min\{m \mid \forall i = 1, 2, \dots, n : \#\{k \leq m \mid X_k = i\} \geq t_i\}, \quad (13)$$

that is, the smallest number of kills such that more than the target number of each item is obtained. Our goal is to find the distribution of $M(t)$, as we did for N in section 1. Thus, we need to compute $P(M(t)=m)$.

We begin by introducing the vector-valued variable $C^m = (C_0^m, C_1^m, \dots, C_n^m)$, where

$$C_i^m = \#\{k \leq m \mid X_k = i\}. \quad (14)$$

That is, C_i^m is the number of times drop i was received after m kills.

Now we consider two cases, as we did for single-item drops. Suppose that we get every drop desired in the minimum number of kills possible, that is, $M(t) = |t|$. There are $\frac{|t|!}{t_1!t_2!\dots t_n!}$ disjoint ways to choose which drops are gotten on which kill⁵, each of which has probability $\prod_{i=1}^n p_i^{t_i}$ by independence and identical distributions of $\{X_k\}$; thus,

$$P(M(t) = |t|) = \frac{|t|!}{t_1!t_2!\dots t_n!} \prod_{i=1}^n p_i^{t_i}. \quad (15)$$

Now suppose that $m > |t|$. Then $M(t) = m$ if and only if the last drop received was the last one needed to meet the target; that is, there is some i such

⁴Additionally, the more general target t can be used to handle certain collection log sections that do not quite fit our model. For example, the number of kills to obtain the ring upgrade component dropped by the Desert Treasure II bosses is negative binomially distributed, not geometrically distributed, so it does not fit our model exactly. Nevertheless, because the number of kills required to obtain r items, each of which requires a geometrically-distributed number of kills to obtain, is negative binomially distributed, it follows that choosing a target of r instead of 1 produces the correct distribution.

⁵There are $|t|!$ ways to rearrange all the drops, which over-counts because it includes cases which differ only by a rearrangement of identical drops, which does not constitute a separate way of receiving the target drops. Thus, we divide by the number of ways to rearrange all the identical drops, which is $t_1!t_2!\dots t_n!$.

that on kill $m - 1$ all but the last of drop i have been received, or $C_{i'}^{m-1} \geq t_{i'}$ if $i' \neq i$, and $C_i^{m-1} = t_i - 1$, and on kill m the last drop i is received, so $X_m = i$. By independence of $\{X_k\}$, we know that $\{X_m=i\}$ is independent of the event $\{C_i^{m-1}=t_i-1, C_{i'}^{m-1} \geq t_{i'} \text{ if } i' \neq i\}$. The choice of different i values leads to disjoint events, so

$$P(M(t)=m) = \sum_{i=1}^n P(X_m=i) P(C_i^{m-1}=t_i-1, C_{i'}^{m-1} \geq t_{i'} \text{ if } i' \neq i).$$

Of course, $P(X_m=i) = p_i$ by assumption, so we focus on the second probability. We note that if $t_i = 0$, then the second probability is 0 as well.

Assume that $t_i \geq 1$. The independence and identical distributions of $\{X_k\}$ imply that C^{m-1} is multinomially distributed⁶ with parameters $m-1$ and (p_0, p_1, \dots, p_n) . Then the distribution of C^{m-1} is

$$P(C^{m-1}=c) = \frac{(m-1)!}{c_0!c_1! \dots c_n!} \prod_{i=0}^n p_i^{c_i}, \quad (16)$$

where $c = (c_0, c_1, \dots, c_n)$ is a vector of nonnegative integers that satisfies

$$|c| = c_0 + c_1 + \dots + c_n = m - 1.$$

Hence,

$$P(C_i^{m-1}=t_i-1, C_{i'}^{m-1} \geq t_{i'} \text{ if } i' \neq i) = \sum_{c \in \mathcal{V}_{t,i}^m} \frac{(m-1)!}{c_0!c_1! \dots c_n!} \prod_{j=0}^n p_j^{c_j}, \quad (17)$$

where

$$\mathcal{V}_{t,i}^m = \{c=(c_0, c_1, \dots, c_n) \mid c_i=t_i-1, c_{i'} \geq t_{i'} \text{ if } i' \neq i, |c|=m-1\}, \quad (18)$$

that is, the set of all possible values of C^{m-1} such that all target drops except for one of drop i have been obtained on kill $m - 1$. Note that $\mathcal{V}_{t,i}^m = \emptyset$ if $t_i = 0$, so equation (17) is still valid even if we remove the assumption $t_i \geq 1$ (under the usual convention that the sum over an empty index set is 0). Thus, if $m > |t|$, we have

$$P(M(t)=m) = \sum_{i=1}^n p_i \sum_{c \in \mathcal{V}_{t,i}^m} \frac{(m-1)!}{c_0!c_1! \dots c_n!} \prod_{j=0}^n p_j^{c_j}. \quad (19)$$

Just as it happened in the single-item case, this formula happens to be the same as the one derived for the case $m = |t|$; although, demonstrating this is a somewhat

⁶One can apply the same reasoning we did for the case $m = |t|$, as that case also concerned a multinomial distribution.

more difficult. Indeed, if $m = |t|$, then the right-hand side of equation (19) becomes

$$\text{RHS}(19) = \sum_{i=1}^n p_i \sum_{c \in \mathcal{V}_{t,i}^{|t|}} \frac{(|t| - 1)!}{c_0! c_1! \cdots c_n!} \prod_{j=0}^n p_j^{c_j}.$$

There is only one element of $\mathcal{V}_{t,i}^{|t|}$, namely $c = (c_0, c_1, \dots, c_n)$, where $c_0 = 0$ (there cannot be any 0 drops if we get all desired drops in the minimum possible number of kills), $c_i = t_i - 1$ (as it always must), and $c_{i'} = t_{i'}$ if $i' \neq i$ (the condition $|c| = m - 1 = |t| - 1$ combined with $c_{i'} \geq t_{i'}$ if $i' \neq i$ leaves only the possibility that $c_{i'} = t_{i'}$). Substituting in gives

$$\begin{aligned} \text{RHS}(19) &= \sum_{i=1}^n p_i \frac{(|t| - 1)!}{t_1! t_2! \cdots (t_i - 1)! \cdots t_n!} p_i^{t_i - 1} \prod_{\substack{j=1 \\ j \neq i}}^n p_j^{t_j} \\ &= \sum_{i=1}^n \frac{t_i (|t| - 1)!}{t_1! t_2! \cdots t_n!} \prod_{j=1}^n p_j^{t_j} \\ &= \left[\frac{(|t| - 1)!}{t_1! t_2! \cdots t_n!} \prod_{j=1}^n p_j^{t_j} \right] \cdot \sum_{i=1}^n t_i \\ &= \frac{|t|!}{t_1! t_2! \cdots t_n!} \prod_{i=1}^n p_i^{t_i} = P(M(t)=m) \end{aligned}$$

by (15). Finally, if we allow $|t| = 0$, then $P(M(t)=m) = 1$ for any $m \geq 1$ (note that $P(M(t)=m) = 0$ if $m < |t|$ or $m \leq 0$, as both outcomes $M(t) < |t|$ and $M(t) \leq 0$ are impossible by the definition of $M(t)$). In this case, formula (19) does not work because there cannot be any element $c \in \mathcal{V}_{t,i}^m$ due to the condition $c_i = t_i - 1 = -1$ conflicting with the requirement that c_i be nonnegative. Empty $\mathcal{V}_{t,i}^m$ implies that (19) would result in 0 for $P(M(t)=m)$ by convention, which is wrong.

So, for any reasonable problem, (19) answers our original question “how long does it take to complete a section of the collection log?” Nevertheless, the formula doesn’t provide any great insights immediately, due mostly to the not-so-simple structure of the set $\mathcal{V}_{t,i}^m$.

Indeed, even directly applying (19) to compute the value $P(M(t)=m)$ for one value of m may be costly, as the number of elements of $\mathcal{V}_{t,i}^m$ can be very large: if $c \in \mathcal{V}_{t,i}^m$, then $c_i = t_i - 1$, but for $i' \neq i$, we can choose $c_{i'}$ freely as long as $c_{i'} \geq t_{i'}$ and $|c| = m$. This is equivalent to saying that, after assigning the first $|t| - 1$ drops so that we $t_i - 1$ of drop i and $t_{i'}$ of each drop i' , we are free to allocate the remaining

$m - |t|$ drops to be any drop we like. By a “bars-and-stars” argument⁷, we get

$$\#\mathcal{V}_{t,i}^m = \binom{m - |t| + n}{n}. \quad (20)$$

Expanding this binomial coefficient using the falling power $x^s = x(x-1)\cdots(x-s+1)$, we have

$$\#\mathcal{V}_{t,i}^m = \frac{(m - |t| + n)^n}{n!} = \mathcal{O}((m - |t|)^n),$$

which does not scale well in m if n is very large at all⁸. Moreover, the product of all p_i values to high powers will be very small in magnitude; if, over the course of the large sum, these small values add up to something significant, then numerical rounding errors could cause a big problem, or else multi-precision methods would be required, further increasing the computational cost.

One idea is to use complementation. Let T^m be the set of all sequences $c = (c_0, c_1, \dots, c_n)$ of nonnegative integers such that $|c| = m - 1$. Then, by another bars-and-stars argument,

$$\#T^m = \binom{m + n}{n} = \frac{(m + n)^n}{n!},$$

so the size of the complement of $\mathcal{V}_{t,i}^m$ is only

$$\begin{aligned} \#T^m - \#\mathcal{V}_{t,i}^m &= \frac{(m + n) \cdots (m + 1) - (m - |t| + n) \cdots (m - |t| + 1)}{n!} \\ &= \mathcal{O}((m - |t|)^{n-1}), \end{aligned}$$

so computing

$$\sum_{c \in \mathcal{V}_{t,i}^m} \frac{(m-1)!}{c_0!c_1!\cdots c_n!} \prod_{j=0}^n p_j^{c_j} = 1 - \sum_{c \in T^m \setminus \mathcal{V}_{t,i}^m} \frac{(m-1)!}{c_0!c_1!\cdots c_n!} \prod_{j=0}^n p_j^{c_j}$$

at least reduces the scaling in m from $\mathcal{O}(m^n)$ down to $\mathcal{O}(m^{n-1})$.

Since most collection log sections have a relatively large number of items ($n = 10$, say), this makes the cost of computing the distribution of $M(t)$ extremely high if a naive approach is used. We need to make some approximations to make practical

⁷The idea of the “bars-and-stars” argument is to think of each bin assignment as a sequence of stars (\star), the items, and bars ($|$), which separate the bins. If you have, say 6 items to assign to 3 bins, then you can do it by generating a sequence of 6 \star ’s and 2 $|$ ’s. For example: $\star\star| \star| \star\star\star$ means assign 2 items to bin 1, 1 item to bin 2, and 3 items to bin 3. This reduces to problem to finding the number of ways to choose 2 slots for the bars out of the 8 available, which is $\binom{6+2}{2}$. In general, if you have L items to assign to B bins, then there are $\binom{L+B-1}{B-1}$ ways to do it.

⁸On the bright side, the cost is at least only polynomial in m , so things could be worse.

computations possible. Perhaps the most important observation we can make is that $0 < p_j < 1$ in general, and usually $p_j \ll 1$ (that is, most drops are very rare), so we expect most of the terms in the inner sum in equation (19) to be extremely small. If these small terms end up contributing a total probability that is still small, that is, if the problem is well-conditioned, then we can simply ignore them, significantly reducing the number of terms we need to compute and add up. This hope is not so unreasonable, as we would be in big trouble no matter what if it turns out to be false.