Appendix: Efficient and Practical Sample Pooling for High-Throughput PCR Diagnosis of COVID-19

Haran Shani-Narkiss, Omri David Gilday, Nadav Yayon, Itamar Daniel Landau*

Center for Brain Sciences, Hebrew University of Jerusalem

* itamar.landau@mail.huji.ac.il

Abstract

In the global effort to combat the COVID-19 pandemic, governments and public health agencies are striving to rapidly increase the volume and rate of diagnostic testing. The most common form of testing today employs Polymerase Chain Reaction in order to identify the presence of viral RNA in individual patient samples one by one. This process has become one of the most significant bottlenecks to increased testing, especially due to reported shortages in the chemical reagents needed in the PCR reaction.

Recent technical advances have enabled High-Throughput PCR, in which multiple samples are pooled into one tube. Such methods can be highly efficient, saving large amounts of time and reagents. However, their efficiency is highly dependent on the frequency of positive samples, which varies significantly across regions and even within regions as testing criterion and conditions change.

Here, we present two possible optimized pooling strategies for diagnostic SARS-CoV-2 testing on large scales, both addressing dynamic conditions. In the first, we employ a simple information-theoretic heuristic to derive a highly efficient re-pooling protocol: an estimate of the target frequency determines the initial pool size, and any subsequent pools found positive are re-pooled at half-size and tested again. In the range of very rare target (<0.05), this approach can reduce the number of necessary tests dramatically, for example, achieving a reduction by a factor of 50 for a target frequency of 0.001. The second method is a simpler approach of optimized one-time pooling followed by individual tests on positive pools. We show that this approach is just as efficient for moderate target-product frequencies ($0.05<0.2$), for example, achieving a two-fold in the number of when the frequency of positive samples is 0.07.

These strategies require little investment, and they offer a significant reduction in the amount of materials, equipment and time needed to test large numbers of samples. We show that both these pooling strategies are roughly comparable to the absolute upper-bound efficiency given by Shannon's source coding theorem. We compare our strategies to the naïve way of testing and to alternative matrix-pooling methods. Most importantly, we offer straightforward, practical pooling instructions for laboratories that perform large scale PCR assays to diagnose SARS-CoV-2 viral particles. These two pooling strategies may offer ways to alleviate the bottleneck currently preventing massive expansion of SARS-CoV-2 testing around the world.

Expected Number of Tests in Repeated Pooling

1 Number of Initial Batches

We assume a large number of samples each with a low probability, p, of testing positive. We look for a protocol of pooling samples into batches that will minimize the expected number of total tests we must perform. We observe that a binary test removes a maximal amount of uncertainty when its two outcomes are equally likely. Therefore the first step of our protocol is to calculate the batch size b such that the entire batch has a probability $\frac{1}{2}$ of testing negative.

We write $q = 1 - p$, for the probability of a single sample testing negative. Then the probability that an entire batch of size b tests negative is q^b, and our desired batch size should satisfy:

$$q^b = \frac{1}{2}$$

Or

$$b = \frac{-\log 2}{\log q} = \log_{\frac{1}{q}} 2$$

Pooling sampes into size $b = \log_{\frac{1}{q}} 2$ yields batches that each have a probability $\frac{1}{2}$ of testing positive/negative.

2 Multiple Positives per Batch

After testing the initial batches of size b, we must further test any positive-resulting batches. If we could be sure each positive batch only had one positive sample in it, then the optimal thing to do would be a binary search: splitting the batch in half, testing one half to identify the batch with a positive sample, and then repeating the process with the positive batch. In our setting there may be more than one positive sample in each batch, and we must therefore test both half-batches and they may both be positive. This could lead to many additional tests if there are many postive samples in any given batch.

How many positive samples do we actually expect in each batch? We compute the conditional probability of finding k positive samples in a batch given that the batch as a whole came up positive. Recall, we have constructed the batches such that each has a probability of $q^b = \frac{1}{2}$ of being entirely negative. Thus the conditional probability distribution on the number of positive samples k in a positive batch is

$$\Pr[k|k > 0] = \frac{\Pr[k]}{\Pr[k > 0]} = 2\Pr[k]$$

for all $k > 0$. And $\Pr[k]$ itself is distributed binomial(b, p) such that we have

$$\Pr[k|k > 0] = 2 \begin{pmatrix} b \\ k \end{pmatrix} p^k q^{b-k} = \begin{pmatrix} b \\ k \end{pmatrix} p^k q^{-k}$$

Specifically, we compute the probability that $k = 1$, given a positive batch.:

$$\Pr[k = 1|k > 0] = \frac{bp}{q} = \frac{p}{q}\log_{\frac{1}{q}} 2$$

The first thing we note is that $\log_{\frac{1}{q}} 2 \approx \frac{\log 2}{p\left(1+\frac{p}{2}\right)}$ to the second order in p. And then to leading order in p we have

$$\Pr[k = 1|k > 0] \approx \frac{\log 2}{(1-p)\left(1+\frac{p}{2}\right)} \approx \left(1+\frac{p}{2}\right)\log 2$$

From this we conclude importantly that the lower bound for $\Pr[k = 1|k > 0]$ is $\log 2 \approx 0.69$. Beyond the lower bound we also verify numerically that the linear approximation holds well for $p < 0.1$. Furthermore, the expected number of positive samples, conditioned on the batch as a whole being positive, is

$$\mathrm{E}[k|k > 0] = 2\mathrm{E}[k] = 2pb = 2p\log_{\frac{1}{q}} 2 \approx 2\left(1-\frac{p}{2}\right)\log 2 < 1.4$$

Therefore, we can conclude that given a positive batch, the probability of it having only 1 positive sample is sufficiently high so as to justify the use of a binary search algorithm (See also the Supplementary Figure, below).

3 Number of Tests on Each Positive Batch

We would like to calculate the expected number of tests that need to be performed on a postive batch of size b, $\mathrm{E}[\text{tests in batch}|b]$. We do this by conditioning on the number of positive samples in the batch. We write $b = 2^n$ for simplicity, and we define

$$N_n^j \equiv \mathrm{E}[\text{tests in batch}|b = 2^n, k = j]$$

Our binary search protocol is as follows: given a positive batch, split it in two, test both halves, and repeat on any positive batches. If we find a positive batch of size $b = 4$ or smaller, we simply test each sample individually because there is no further benefit to splitting the batch again. In order to find one positive sample in a batch of size b we will need to split the batch $\log_2 b$ times and perform two tests after each splitting. Thus if $k = 1$ we perform $2\log_2 b$ tests per positive batch. Thus, we have

$$N_n^1 = 2n$$

We can alternately derive this expression for N_n^1 recursively, and we do that here as an introduction to what follows. Given a batch of size $b = 2^n$ with exactly $k = 1$ positives, by our protocol we will perform 2 tests and then split the batch into two half-batches of size $b = 2^{n-1}$. By our assumption that $k = 1$ we know that exactly one of the half batches will be positive, and therefore require an additional N_{n-1}^1 tests. Thus we have the recursive relation

$$N_n^1 = 2 + N_{n-1}^1$$

The stopping condition of the recursion is $N_2^1 = 4$. So unrolling this expression we have

$$N_n^1 = \underbrace{2 + 2 + ... + 2}_{2(n-2)} + 4 = 2n$$

Now suppose there are $k = 2$ positives in a batch . Splitting the batch in two is analogous to flipping two coins. There is a $\frac{1}{2}$ chance the two positive samples will be in distinct half-batches, in which case we perform two binary searches on batches of size $b = 2^{n-1}$ each with $k = 1$, i.e. we perform an additional $N_{n-1}^1 = 2(n-1)$ tests. There is also a $\frac{1}{2}$ chance that both positive samples will be in the same half-batch, in which case we will need to again perform binary search on a batch with $k = 2$, this time of size $b = 2^{n-1}$. Therefore we can write a recursive relationship for the number N_n^2:

$$N_n^2 = 2 + N_{n-1}^1 + \frac{1}{2}N_{n-1}^2 = 2n + \frac{1}{2}N_{n-1}^2$$

and this continues until $n = 2$, at which point we simply perform $N_2^2 = 4$ additional tests. We unroll this recursion to find:

$$N_n^2 = 2n + (n-1) + \frac{1}{2}(n-2) + ... + 4$$

We can rewrite this as a summation by observing that each term is a product of a whole number j, that decreases by 1 at each step, multiplied by a power of 2 that decreases by 1 at each step. Thus we write

$$N_n^2 = \sum_{j=2}^{n} 2^{1+j-n} j$$

Next we rewrite this as

$$N_n^2 = 2^{2-n} \sum_{j=2}^{n} 2^{j-1} j$$

and observe that $\sum_j r^{j-1} j = \frac{\partial}{\partial r} \sum_j r^j$. Furthermore,

$$\sum_{j=2}^{n} r^j = \sum_{j=0}^{n} -r - 1 = \frac{1 - r^{n+1}}{1 - r} - r - 1$$

Thus by calculating the necessary derivative we find

$$\sum_{j=2}^{n} 2^{j-1} j = \frac{\partial}{\partial r} \left(\frac{1 - r^{n+1}}{1 - r} \right)_{r=2} - 1 = 2^n (n-1)$$

Thus finally we have:

$$N_n^2 = 4(n-1)$$

That is, the expected number of tests needed for a batch with two positive samples is $4(\log_2 b - 1)$.

Now we would like to calculate

$$\mathrm{E}[\text{tests in batch} | b = 2^n] = \sum_{k=1}^{\infty} N_n^k \Pr[k | k > 0]$$

For $k = 1$ recall the approximation derived above: $\Pr[k = 1 | k > 0] \approx \left(1 + \frac{p}{2}\right) \log 2$. In practice we find that the p-dependence does not impact the results of this calculation within the range of $p < 0.2$, and we therefore use only the zeroth order term:

$$\Pr[k = 1 | k > 0] \approx \log 2$$

We observe numerically that the probability of a positive batch having more than two positive samples is smaller than 0.07 for all p, and therefore we approximate the number of tests per positive batch by assuming a positive batch has either 1 or 2 positive samples (Supplementary Figure). Thus we write

$$\Pr\left[k = 2 | k > 0\right] \approx 1 - \log 2$$

And therefore we approximate

$$\mathrm{E}\left[\text{tests in batch} | b = 2^n\right] \approx 2n \log 2 + 4\left(n - 1\right)\left(1 - \log 2\right)$$

or

$$\mathrm{E}\left[\text{tests in batch} | b = 2^n\right] \approx 2n\left(2 - \log 2\right) - 4\left(1 - \log 2\right)$$

4 Expected Number of Total Tests

If we begin with N total samples, then the number of initial batches to test is $\frac{N}{b}$. By construction the expected number of positive batches is one half of that, so the expected total number of tests is

$$N_{tests} = \frac{N}{b}\left(1 + \frac{1}{2}\mathrm{E}\left[\text{tests in batch} | k > 0\right]\right)$$

Using our approximation for $\mathrm{E}\left[\text{tests in batch} | b = 2^n\right]$ above, with $n = \log_2 b$, we have

$$N_{tests} \approx \frac{N}{b}\left(\underbrace{\left(2 - \log 2\right)}_{\approx 1.3}\log_2 b + \underbrace{2\log 2 - 1}_{\approx 0.4}\right)$$

as presented in the main text.

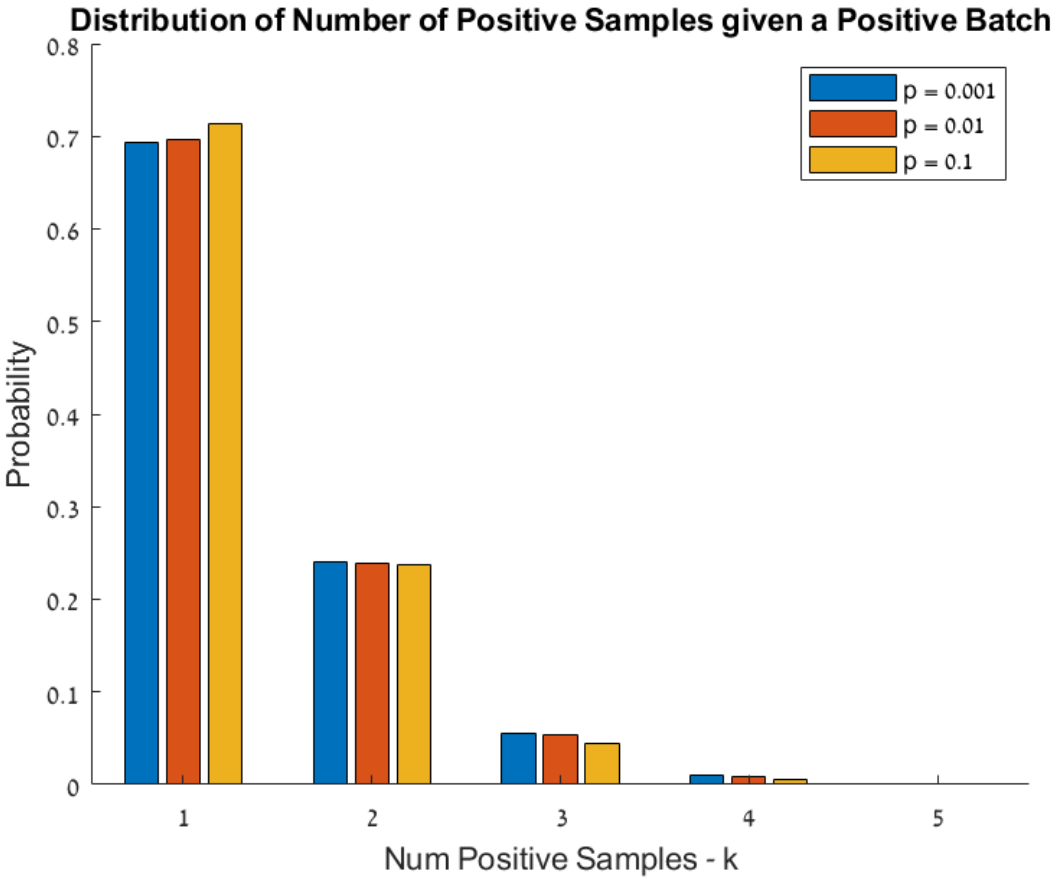

Figure 1: **Conditional Distribution of the Number of Positive Samples Given a Positive Batch for Different Values of p**