Achieving maximum speedup in multi-level acceleration for massive coronavirus testing

Keqin Li & Bo Yang

To cite this article: Keqin Li & Bo Yang (2023): Achieving maximum speedup in multi-level acceleration for massive coronavirus testing, International Journal of Parallel, Emergent and Distributed Systems, DOI: 10.1080/17445760.2023.2190975

To link to this article: https://doi.org/10.1080/17445760.2023.2190975

Published online: 25 Mar 2023.

Submit your article to this journal

View related articles

View Crossmark data
Achieving maximum speedup in multi-level acceleration for massive coronavirus testing

Keqin Li and Bo Yang

ABSTRACT
It is well and widely known that sample pooling could provide an effective and efficient way for fast coronavirus testing among massive asymptomatic individuals. The method of multi-level acceleration for asymptomatic COVID-19 screening has been introduced, and for one and two levels, the optimal group sizes have been obtained. However, there are still multiple challenges. First, it is not clear how to find the optimal group sizes for three or more levels. Second, there is lack of closed-form expressions for the optimal group sizes for two or more levels. Third, it is not clear how to determine the optimal number of levels. And last, it is not known what the maximum achievable speedup is. The motivation of this paper is to address all the above challenges. The optimization of a hierarchical pooling strategy includes its number of levels and the group size of each level. In this paper, based on multi-variable optimization and Taylor approximation, we are able to derive closed-form expressions for the optimal number of levels \( d^* = \ln\left(\frac{1}{\ln\left(\frac{1}{p_0}\right)}\right) - 1 \), the optimal group sizes \( m_1^* = \frac{e}{e^{d^*}} = 1/(e p_0) \), \( m_2^* = \frac{e^{d^* - 1}}{e^2 p_0} \), ..., \( m_d^* = \frac{e}{e^{d^*}} = 1/(e^{d^*} p_0) \), and the maximum possible speedup of a hierarchical pooling strategy of \( 1/(e p_0 \ln\left(\frac{1}{p_0}\right)) \), where \( p_0 \) is the fraction of infected people. The above speedup is nearly a linear function of the reciprocal of \( p_0 \), in the sense that it is asymptotically greater than any sub-linear function \( (1/p_0)^{1-\epsilon} \) of the reciprocal of \( p_0 \) for any small \( \epsilon > 0 \). Using the results in this paper, we can quickly and easily predict the performance of an optimal hierarchical pooling strategy. For instance, if the fraction of infected people is 0.0001, an 8-level hierarchical pooling strategy can achieve speedup of nearly 400.

1. Introduction
1.1. Background

It is well and widely known that sample pooling could provide an effective and efficient way for fast coronavirus testing among massive asymptomatic individuals [1,2]. Sample pooling strategies can save substantial time and resources compared to individual testing during epidemic surveillance and large-scale COVID-19 screening [3,4]. It was reported that up to 89% fewer tests would be required for group size of 3–25 in a population of 150,000 with an infection prevalence of 1% [5]. It was also found that by pooling 384 samples into 48 groups, both an 8-fold increase in testing efficiency and an...
8-fold reduction in test costs can be achieved [6]. The approach of sample pooling and group testing has been introduced [7,8], adopted and applied [9–15], extensively studied [5,6,16–22], and reviewed [23,24].

The method of multi-level acceleration for asymptomatic COVID-19 screening has been introduced in [25]. For one and two levels, the optimal group sizes were obtained in [25]. However, there are still multiple challenges. First, it is not clear how to find the optimal group sizes for three or more levels. Second, there is lack of closed-form expressions for the optimal group sizes for two or more levels. Third, it is not clear how to determine the optimal number of levels. And last, it is not known what the maximum achievable speedup is. The motivation of this paper is to address all the above challenges.

1.2. Contributions

The optimization of a hierarchical pooling strategy includes its number of levels and the group size of each level. In this paper, based on multi-variable optimization and Taylor approximation, we are able to derive closed-form expressions for the optimal number of levels \( d^* = \ln(1/\ln(1/q_0)) - 1 \), the optimal group sizes \( m_1^* = ee^{d^*} = 1/(ep_0), m_2^* = e^{d^* - 1} = 1/(e^2p_0), ..., m_d^* = e = 1/(e^d p_0) \), and the maximum possible speedup of a hierarchical pooling strategy of \( 1/(ep_0 \ln(1/p_0)) \), where \( p_0 \) is the fraction of infected people. The above speedup is nearly a linear function of the reciprocal of \( p_0 \), in the sense that it is asymptotically greater than any sub-linear function \( (1/p_0)^{1-\epsilon} \) of the reciprocal of \( p_0 \) for any small \( \epsilon > 0 \).

The paper is organized as follows. In Section 2, we describe the hierarchical pooling strategy and analyze its performance. In Section 3, we derive closed-form expressions for the optimal group sizes for one and two levels. We confirm their accuracy by comparing them with known solutions. In Section 4, we derive closed-form expressions for the optimal group sizes and the optimal number of levels. We also demonstrate numerical data. We conclude the paper in Section 5.

2. Hierarchical pooling strategy

In this section, we describe the hierarchical pooling strategy and analyze its performance.

2.1. Description of the strategy

A hierarchical pooling strategy involves pooling samples from multiple people and works as follows. A \( d \)-level hierarchical pooling strategy (HPS\(_d\)) has \( d \geq 1 \) levels. The size of a level-\( j \) group is \( m_j \), where \( 1 \leq j \leq d \). For convenience, a population of size \( N \) can be treated as a level-0 group of size \( m_0 = N \). A level-\( j \) group is divided into level-(\( j + 1 \)) groups of size \( m_{j+1} \), where \( 0 \leq j \leq d - 1 \). A level-\( d \) group cannot be further divided. It is clear that \( m_0 > m_1 > m_2 > \cdots > m_d > 1 \).

\[
\text{Algorithm 1: HPS}_d(j, S)
\]

\textbf{Input:} A level \( j, 1 \leq j \leq d \); a set \( S \) of \( m_j \) samples.

\textbf{Output:} A subset \( P \subseteq S \) of positive samples.

\[
P \leftarrow \emptyset; \quad (1)
\]

Perform a group test for \( S \); \quad (2)

\textbf{if} (the group test result of \( S \) is negative)

\textbf{return} \( P \); \quad (3)

\textbf{end if}; \quad (4)

\textbf{if} \( (j < d) \)

\( n \leftarrow \lfloor m_j / m_{j+1} \rfloor \); \quad (5)

Divide \( S \) into \( S_1, S_2, ..., S_n \); \quad (6)

...
Algorithm 1 gives a recursive description of the HPS$_d$ procedure. On level $j$, a group test is performed for a level-$j$ group (which is divided from a level-$(j-1)$ group) of size $m_j$ (line 2). If the test result of a level-$j$ group of $m_j$ samples is negative, we know that all the individual samples in the group are negative (lines 3–5). If the test result of a level-$j$ group of $m_j$ samples is positive, where $1 \leq j \leq d - 1$, then the $m_j$ samples proceed to level $j + 1$, i.e. they are divided into level-$(j+1)$ groups of size $m_{j+1}$, which are processed by using the same HPS$_d$ procedure (lines 7–12). One level $d$, the individual samples of a level-$d$ group are tested one by one without sample pooling (lines 14–19).

**2.2. Analysis of the strategy**

Let us define the following variables.

- $p_0$: the probability that the test result of one individual is positive.
- $q_0$: the probability that the test result of one individual is negative.
- $p_j$: the probability that the test result of one level-$j$ group is positive under the condition that the test result of a level-$(j-1)$ group is positive, where $1 \leq j \leq d$.
- $q_j$: the probability that the test result of one level-$j$ group is negative under the condition that the test result of a level-$(j-1)$ group is positive, where $1 \leq j \leq d$.
- $T_j$: the expected number of tests for one level-$j$ group, where $1 \leq j \leq d$.
- $T'_j$: the expected number of tests for one level-$j$ group under the condition that the test result of the level-$j$ group is positive, where $1 \leq j \leq d$.

The following theorem gives $p_j$ and $q_j$ for all $1 \leq j \leq d$.

**Theorem 2.1:** For a d-level hierarchical pooling strategy, we have $q_1 = q_0^{m_1}$, $p_1 = 1 - q_1$, and

\[
q_j = \frac{q_0^{m_j} - q_0^{m_{j-1}}}{p_1 p_2 \cdots p_{j-1}},
\]

and $p_j = 1 - q_j$, for all $2 \leq j \leq d$.

**Proof:** The equations for $q_1$ and $p_1$ are straightforward. As for $q_j$, where $2 \leq j \leq d$, we have

\[
q_j = \frac{q_0^{m_j} (1 - q_0^{m_{j-1}-m_j})}{p_1 p_2 \cdots p_{j-1}} = \frac{q_0^{m_j} - q_0^{m_{j-1}}}{p_1 p_2 \cdots p_{j-1}}.
\]
where $p_1 p_2 \cdots p_{j-1}$ is the probability that the test result of a level-$(j-1)$ group is positive (i.e. the condition), which implies that the test results of all corresponding level-1,..., level-$(j-2)$ groups are positive; $q_0^{m_j}$ is the probability that all the $m$ samples in a level-j group are negative (i.e. the test result of one level-j group is negative); and $(1 - q_0^{m_j-1})$ is the probability that at least one of the remaining $(m_j - m_i)$ samples in the same level-$(j-1)$ group is positive (to keep the condition). The equations for $p_j$, where $2 \leq j \leq d$, are straightforward.

The following theorem gives closed-from expressions of $p_j$ and $q_j$ for all $2 \leq j \leq d$.

**Theorem 2.2:** For a $d$-level hierarchical pooling strategy, we have

$$p_j = \frac{1 - q_0^{m_j}}{1 - q_0^{m_{j-1}}},$$

and

$$q_j = \frac{q_0^{m_j} - q_0^{m_{j-1}}}{1 - q_0^{m_{j-1}}},$$

for all $2 \leq j \leq d$.

**Proof:** We can prove by induction on $j \geq 2$. First, it is easy to verify that the claim is correct for $p_2$ and $q_2$. Next, we assume that the claim holds for $p_2$ and $q_2$, ..., $p_{j-1}$ and $q_{j-1}$. For $q_j$, we notice that

$$p_1 p_2 \cdots p_{j-1} = (1 - q_0^{m_1}) \left( \frac{1 - q_0^{m_2}}{1 - q_0^{m_1}} \right) \cdots \left( \frac{1 - q_0^{m_{j-1}}}{1 - q_0^{m_{j-2}}} \right) = 1 - q_0^{m_{j-1}},$$

by the induction hypothesis, which yields $q_j$ and $p_j$. ■

Let $T_{\text{pooling}}(m_1, m_2, \ldots, m_d)$ be the expected number of tests of a $d$-level hierarchical pooling strategy. The following theorem gives $T_{\text{pooling}}(m_1, m_2, \ldots, m_d)$, and $T_j$ and $T'_j$ for all $1 \leq j \leq d$.

**Theorem 2.3:** For a $d$-level hierarchical pooling strategy, we have

$$T_{\text{pooling}}(m_1, m_2, \ldots, m_d) = \left( \frac{N}{m_1} \right) T_1,$$

$$T_j = q_j + (T'_j + 1)p_j = 1 + p_j T'_j, \quad 1 \leq j \leq d,$$

$$T'_j = \left( \frac{m_j}{m_{j+1}} \right) T_{j+1}, \quad 1 \leq j \leq d - 1,$$

$$T'_d = m_d.$$

**Proof:** The equation for $T_{\text{pooling}}(m_1, m_2, \ldots, m_d)$ is straightforward. For a level-$j$ group of samples, if the test result of the group is negative (which happens with probability $q_j$), only one test is required; if the test result of the group is positive (which happens with probability $p_j$), $T'_j + 1$ tests are required, one for group test, and $T'_j$ for proceeding to level $j+1$. Hence, the expected number of tests for one level-$j$ group is $T_j = q_j + (T'_j + 1)p_j = 1 + p_j T'_j$, for all $1 \leq j \leq d$. The equation for $T'_j$ is straightforward for all $1 \leq j \leq d$. ■

The following theorem gives a closed-from expression of $T_j$ for all $1 \leq j \leq d$. 


**Theorem 2.4:** For a d-level hierarchical pooling strategy, we have

\[
T_j = 1 + m_j \left( \frac{p_j}{m_{j+1}} + \frac{p_j p_{j+1}}{m_{j+2}} + \cdots + \frac{p_j p_{j+1} \cdots p_{d-1}}{m_d} + p_j p_{j+1} \cdots p_d \right),
\]

for all \(1 \leq j \leq d\).

**Proof:** We can prove by induction on \(j = d, d - 1, \ldots, 1\). First, it is easy to verify that

\[
T_d = 1 + p_d T'_d = 1 + m_d p_d.
\]

Next, we assume that the claim holds for \(T_{j+1}\). For \(T_j\), we have

\[
T_j = 1 + p_j T'_j = 1 + p_j \left( \frac{m_j}{m_{j+1}} \right) T_{j+1} = 1 + p_j \left( \frac{m_j}{m_{j+1}} \right) \left( 1 + m_{j+1} \left( \frac{p_{j+1}}{m_{j+2}} + \cdots + \frac{p_{j+1} p_{j+2} \cdots p_{d-1}}{m_d} + p_{j+1} p_{j+2} \cdots p_d \right) \right).
\]

This proves the theorem.

Note that the number of tests without sample pooling is \(N\). Therefore, the speedup of a d-level hierarchical pooling strategy is

\[
S(m_1, m_2, \ldots, m_d) = \frac{N}{T_{\text{pooling}}(m_1, m_2, \ldots, m_d)} = \frac{m_1}{T_1}.
\]

The biggest challenge is to find \(m_1, m_2, \ldots, m_d\), such that \(S(m_1, m_2, \ldots, m_d)\) is maximized. In fact, the number \(d\) of levels should also be optimized.

## 3. Closed-form expressions

In this section, we derive closed-form expressions for the optimal group sizes when \(d = 1\) and \(d = 2\).

The key method to derive closed-form expressions is to use the following approximation. For the function \(f(x) = \ln x\), we use the Taylor approximation \(f(x) = f(1) + f'(1)(x - 1)\) at 1, that is, \(\ln x = x - 1\), or \(x = \ln x + 1\), for \(x \approx 1\). Letting \(x = q_0^b\), we get

\[
q_0^b = k \ln q_0 + 1 = 1 - k \ln(1/q_0).
\]

The above equation is repeatedly used in this paper.

### 3.1. One-level acceleration

The following theorem gives closed-form expressions of the optimal group size and the maximum speedup when \(d = 1\).

**Theorem 3.1:** When \(d = 1\), the optimal group size is

\[
m_1^* = \sqrt{\frac{1}{\ln(1/q_0)}}.
\]

The speedup achieved is

\[
m_1^*/2 = \frac{1}{2} \sqrt{\frac{1}{\ln(1/q_0)}}.
\]
Table 1. Optimal group size for one-level pooling strategy.

<table>
<thead>
<tr>
<th>$p_0$</th>
<th>$m_1^*$ from [25]</th>
<th>$m_1^*$ (closed-form)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>101</td>
<td>100</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>317</td>
<td>316</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>1001</td>
<td>1000</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>3163</td>
<td>3162</td>
</tr>
</tbody>
</table>

Proof: For a one-level pooling strategy with group size $m_1$, we have

$$T_1 = 1 + m_1 p_1 = 1 + m_1 (1 - q_0^{m_1}),$$

and

$$S(m_1) = \frac{m_1}{T_1} = \frac{1}{1 + 1/m_1 - q_0^{m_1}}.$$  

To find the optimal value of $m_1$, we need to minimize

$$F(m_1) = \frac{1}{m_1} - q_0^{m_1} = \frac{1}{m_1} - 1 + m_1 \ln(1/q_0).$$

Note that

$$\frac{\partial F(m_1)}{\partial m_1} = -\frac{1}{m_1^2} + \ln(1/q_0) = 0,$$

which gives the optimal group size $m_1^*$ as

$$m_1^* = \sqrt{\frac{1}{\ln(1/q_0)}}.$$  

Furthermore, we have the optimal speedup

$$S(m_1^*) = \frac{1}{1 + 1/m_1^* - q_0^{m_1^*}} = \frac{1}{1/m_1^* + m_1^* \ln(1/q_0)} = \frac{1}{2 \sqrt{\frac{1}{\ln(1/q_0)}}} = \frac{m_1^*}{2}.$$  

This proves the theorem. ■

Table 1 shows the accuracy of the above closed-form expression of $m_1^*$ (actually $\lceil m_1^* \rceil$) compared with the real optimal value of $m_1$ obtained in [25]. It is easily seen that our closed-form expression is very accurate.

3.2. Two-Level acceleration

The following theorem gives closed-form expressions of the optimal group sizes and the maximum speedup when $d = 2$. 
Theorem 3.2: When $d = 2$, the optimal group sizes are

$$m_1^* = \left( \frac{1}{\ln(1/q_0)} \right)^{2/3},$$

and

$$m_2^* = \left( \frac{1}{\ln(1/q_0)} \right)^{1/3}.$$

The speedup achieved is

$$S(m_1) = \frac{m_1}{T_1} = \frac{m_1}{2\sqrt{m_1(1 - q_0^{m_1})} + 1} = \frac{1}{2\sqrt{m_1 \ln(1/q_0) + 1/m_1}}.$$
Table 2. Optimal group sizes for two-level pooling strategy.

<table>
<thead>
<tr>
<th>$p_0$</th>
<th>$(m_1^<em>, m_2^</em>)$ from [25]</th>
<th>$(m_1^<em>, m_2^</em>)$ (closed-form)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}$</td>
<td>(8, 2)</td>
<td>(4, 2)</td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td>(25, 5)</td>
<td>(21, 5)</td>
</tr>
<tr>
<td>$10^{-3}$</td>
<td>(106, 10)</td>
<td>(100, 10)</td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td>(476, 22)</td>
<td>(464, 22)</td>
</tr>
<tr>
<td>$10^{-5}$</td>
<td>(2179, 46)</td>
<td>(2154, 46)</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>(10051, 100)</td>
<td>(10000, 100)</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>(46525, 215)</td>
<td>(46416, 215)</td>
</tr>
</tbody>
</table>

We need to minimize

$$F(m_1) = 2\sqrt{m_1 \ln(1/q_0)} + \frac{1}{m_1}.$$  

Note that

$$\frac{\partial F(m_1)}{\partial m_1} = \frac{\ln(1/q_0)}{\sqrt{m_1}} - \frac{1}{m_1^2} = 0,$$

which gives

$$m_1^* = \left(\frac{1}{\ln(1/q_0)}\right)^{2/3},$$

and

$$m_2^* = \left(\frac{1}{\ln(1/q_0)}\right)^{1/3}.$$

Furthermore, we have

$$S(m_1^*, m_2^*) = \frac{1}{2\sqrt{m_1^* \ln(1/q_0)} + 1/m_1^*} = \frac{1}{3} \left(\frac{1}{\ln(1/q_0)}\right)^{2/3} = \frac{m_1^*}{3}.$$  

This proves the theorem.

Table 2 shows the accuracy of the above closed-form expressions of $m_1^*$ and $m_2^*$ (actually $\lfloor m_1^* \rfloor$ and $\lfloor m_2^* \rfloor$) compared with the real optimal values of $m_1$ and $m_2$ obtained in [25]. It is easily seen that our closed-form expressions are very accurate, especially when $p_0$ is small.

4. Multi-level acceleration

In this section, we derive closed-form expressions for the optimal group sizes and the optimal number of levels for a hierarchical pooling strategy.

The main result of this section is the following theorem, which gives closed-form expressions of the optimal number of levels, the optimal group sizes, and the maximum speedup for all $d \geq 1$.

**Theorem 4.1:** For all $d \geq 1$, the optimal number of levels is

$$d^* = \ln\left(\frac{1}{\ln(1/q_0)}\right) - 1.$$

The optimal group sizes are

$$m_j^* = \left(\frac{1}{\ln(1/q_0)}\right)^{(d^*+1-j)/(d^*+1)} = e^{d^*+1-j} = \frac{1}{e^{j/p_0}}.$$
for all $1 \leq j \leq d^*$. The speedup achieved is

$$\frac{m^*_1}{d^* + 1} = \frac{1}{d^* + 1} \left( \frac{1}{\ln(1/q_0)} \right)^{d^*/(d^*+1)},$$

which is actually

$$\frac{1}{\ln(1/\ln(1/q_0))} \left( \frac{1}{\ln(1/q_0)} \right)^{(\ln(1/\ln(1/q_0)) - 1)/\ln(1/\ln(1/q_0))},$$

or equivalently,

$$\frac{1}{\ln(1/\ln(1/(1-p_0)))} \left( \frac{1}{\ln(1/(1-p_0))} \right)^{(\ln(1/\ln(1/(1-p_0))) - 1)/\ln(1/\ln(1/(1-p_0)))} = \frac{1}{ep_0 \ln(1/p_0)}.$$

The rest of the section is devoted to proving the above theorem.

### 4.1. Optimal group sizes

Now, let us consider a $d$-level hierarchical pooling strategy with group sizes $m_1, m_2, \ldots, m_d$. By Theorem 2.4, we know that

$$T_{\text{pooling}}(m_1, m_2, \ldots, m_d) = N \left( \frac{1}{m_1} + \frac{p_1}{m_2} + \frac{p_1 p_2}{m_3} + \cdots + \frac{p_1 p_2 \cdots p_{d-1}}{m_d} + \frac{p_1 p_2 \cdots p_d}{m} \right),$$

which is actually

$$T_{\text{pooling}}(m_1, m_2, \ldots, m_d) = N \left( \frac{1}{m_1} + \frac{1 - q_0^{m_1}}{m_2} + \frac{1 - q_0^{m_2}}{m_3} + \cdots + \frac{1 - q_0^{m_{d-1}}}{m_d} + (1 - q_0^{m_d}) \right),$$

and approximately,

$$T_{\text{pooling}}(m_1, m_2, \ldots, m_d) = N \left( \frac{1}{m_1} + \ln(1/q_0) \left( \frac{m_1}{m_2} + \frac{m_2}{m_3} + \cdots + \frac{m_{d-1}}{m_d} + m_d \right) \right).$$

The above approximation makes it possible to derive the optimal group sizes in closed-form. To minimize $T_{\text{pooling}}(m_1, m_2, \ldots, m_d)$, we need to minimize

$$F(m_1, m_2, \ldots, m_d) = \frac{1}{m_1} + \ln(1/q_0) \left( \frac{m_1}{m_2} + \frac{m_2}{m_3} + \cdots + \frac{m_{d-1}}{m_d} + m_d \right).$$

This requires

$$\frac{\partial F(m_1, m_2, \ldots, m_d)}{\partial m_1} = -\frac{1}{m_1} + \frac{\ln(1/q_0)}{m_2} = 0,$$

$$\frac{\partial F(m_1, m_2, \ldots, m_d)}{\partial m_2} = \ln(1/q_0) \left( -\frac{m_1}{m_2^2} + \frac{1}{m_3} \right) = 0,$$

$$\frac{\partial F(m_1, m_2, \ldots, m_d)}{\partial m_3} = \ln(1/q_0) \left( -\frac{m_2}{m_3^2} + \frac{1}{m_4} \right) = 0,$$

$$\vdots$$

$$\frac{\partial F(m_1, m_2, \ldots, m_d)}{\partial m_{d-1}} = \ln(1/q_0) \left( -\frac{m_{d-2}}{m_{d-1}^2} + \frac{1}{m_d} \right) = 0.$$
\[
\frac{\partial F(m_1, m_2, \ldots, m_d)}{\partial m_d} = \ln(1/q_0) \left( -\frac{m_{d-1}}{m_d} + 1 \right) = 0.
\]

Solving the above equations, we get

\[
m_d^* = (m_{d-1}^*)^{1/2} = \left( \frac{1}{\ln(1/q_0)} \right)^{1/(d+1)},
\]
\[
m_{d-1}^* = (m_{d-2}^*)^{2/3} = \left( \frac{1}{\ln(1/q_0)} \right)^{2/(d+1)},
\]
\[
m_{d-2}^* = (m_{d-3}^*)^{3/4} = \left( \frac{1}{\ln(1/q_0)} \right)^{3/(d+1)},
\]
\[
\vdots
\]
\[
m_2^* = (m_1^*)^{(d-1)/d} = \left( \frac{1}{\ln(1/q_0)} \right)^{(d-1)/(d+1)},
\]
\[
m_1^* = \left( \frac{1}{\ln(1/q_0)} \right)^{d/(d+1)},
\]

which give

\[
T_{\text{pooling}}(m_1^*, m_2^*, \ldots, m_d^*) = N \left( (\ln(1/q_0))^{d/(d+1)} + \ln(1/q_0) \left( \frac{1}{\ln(1/q_0)} \right)^{1/(d+1)} \right) = N \left( \frac{d+1}{m_1^*} \right),
\]

and the speedup is

\[
S(m_1^*, m_2^*, \ldots, m_d^*) = \frac{N}{T_{\text{pooling}}(m_1^*, m_2^*, \ldots, m_d^*)} = \frac{m_1^*}{d+1} = \frac{1}{d+1} \left( \frac{1}{\ln(1/q_0)} \right)^{d/(d+1)}.
\]

### 4.2. Optimal number of levels

To find the optimal number of levels, we view the speedup as a function of \(d\):

\[
S(d) = \frac{1}{d+1} \left( \frac{1}{\ln(1/q_0)} \right)^{d/(d+1)}.
\]

To maximize \(S(d)\), we need \(\partial S(d) / \partial d = 0\), where

\[
\frac{\partial S(d)}{\partial d} = \frac{1}{(d+1)^2} \left( \frac{1}{\ln(1/q_0)} \right)^{d/(d+1)} \left( \frac{1}{d+1} \ln \left( \frac{1}{\ln(1/q_0)} \right) - 1 \right),
\]

which gives the optimal number of levels \(d^*\) as

\[
d^* = \ln \left( \frac{1}{\ln(1/q_0)} \right) - 1.
\]
Algorithm 2: HPS Optimization

**Input:** $p_0$.
**Output:** $d^*, m^*_1, m^*_2, \ldots, m^*_d$.

1. Calculate $d^* = \left\lfloor \ln(1/\ln(1/(1 - p_0))) - 1 \right\rfloor$; (1)
2. for $j \leftarrow 1$ to $d^*$ do
   3. Calculate $m^*_j = \left\lfloor (1/\ln(1/(1 - p_0)))^{(d^*+1-j)/(d^*+1)} \right\rfloor$; (3)
   4. end for; (4)
4. return $d^*, m^*_1, m^*_2, \ldots, m^*_d$. (5)

Algorithm 2 gives our method to find the optimal hierarchical pooling strategy with the optimal number of levels and the optimal group sizes.

### 4.3. The maximum speedup

The maximum achievable speedup of a hierarchical pooling strategy is a function of $q_0$:

$$S(q_0) = \frac{1}{\ln(1/\ln(1/q_0))} \left( \frac{1}{\ln(1/q_0)} \right)^{(\ln(1/\ln(1/q_0))-1)/\ln(1/(1/q_0))},$$

or equivalently, a function of $p_0$:

$$S(p_0) = \frac{1}{\ln(1/\ln(1/(1 - p_0)))} \left( \frac{1}{\ln(1/(1 - p_0))} \right)^{(\ln(1/\ln(1/(1-p_0)))-1)/\ln(1/(1-p_0))}. $$

To simplify the above expression, let

$$x = \frac{1}{\ln(1/q_0)} = \frac{1}{\ln(1/(1 - p_0))}.$$

Then, we get $d^* = \ln x - 1$, and

$$S(p_0) = \frac{x^{1-1/\ln x}}{\ln x} = \frac{x}{(\ln x)^{1/\ln x}}.$$

Notice that $x^{1/\ln x} = e$. Hence, we get

$$S(p_0) = \frac{x}{e \ln x}.$$

Since

$$\frac{1}{1 - p_0} = 1 + p_0 + p_0^2 + \cdots = 1 + p_0 + o(p_0),$$

and

$$\ln(1 + y) = y - \frac{y^2}{2} + \frac{y^3}{3} - \cdots,$$

we have (by setting $y = p_0 + o(p_0)$)

$$\ln \left( \frac{1}{1 - p_0} \right) = (p_0 + o(p_0)) - \frac{1}{2} (p_0 + o(p_0))^2 + \frac{1}{3} (p_0 + o(p_0))^3 - \cdots = p_0 + o(p_0),$$
Figure 1. Speedup vs. number of levels \( (p_0 = 0.001) \).

Table 3. Optimal number of levels, optimal group sizes, and maximum speedup.

<table>
<thead>
<tr>
<th>( p_0 )</th>
<th>( d^* )</th>
<th>( {m_1^<em>, m_2^</em>, \ldots, m_d^*} )</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 10^{-1} )</td>
<td>1</td>
<td>(3)</td>
<td>1.55</td>
</tr>
<tr>
<td>( 10^{-2} )</td>
<td>4</td>
<td>(37, 13, 5, 2)</td>
<td>7.96</td>
</tr>
<tr>
<td>( 10^{-3} )</td>
<td>6</td>
<td>(368, 135, 50, 18, 7, 2)</td>
<td>53.23</td>
</tr>
<tr>
<td>( 10^{-4} )</td>
<td>8</td>
<td>(3679, 1353, 498, 183, 67, 25, 9, 3)</td>
<td>399.40</td>
</tr>
<tr>
<td>( 10^{-5} )</td>
<td>11</td>
<td>(36788, 13533, 4979, 1832, 674, 248, 91, 34, 12, 5, 2)</td>
<td>3195.35</td>
</tr>
<tr>
<td>( 10^{-6} )</td>
<td>13</td>
<td>(367879, 135335, 49787, 18316, 6738, 2479, 912, 335, 123, 45, 17, 6, 2)</td>
<td>26627.99</td>
</tr>
<tr>
<td>( 10^{-7} )</td>
<td>15</td>
<td>(3678794, 1353353, 497871, 183156, 67379, 24788, 9119, 3355, 1234, 454, 167, 61, 23, 8, 3)</td>
<td>228240.01</td>
</tr>
</tbody>
</table>

and \( x = 1/p_0 \). Therefore, we obtain

\[
S(p_0) = \frac{1}{e^{p_0} \ln(1/p_0)}.
\]

By using the above technique, we can have \( m_{d^*}^* = e = 1/(e^{d^*}p_0) \), \( m_{d^*-1}^* = e^2 = 1/(e^{d^*-1}p_0) \), \( m_{d^*-2}^* = e^3 = 1/(e^{d^*-2}p_0) \), \ldots, \( m_1^* = e^{d^*} = 1/(ep_0) \).

We have proved Theorem 4.1.

4.4. Numerical data

We now demonstrate some numerical data.

In Figure 1, for \( p_0 = 0.001 \), we show the speedup \( S(d) \) as a function of number of levels \( d \). It can be observed that as \( d \) increases, \( S(d) \) also increases. However, to certain point, \( S(d) \) decreases as \( d \) further increases. It is clear that there is an optimal value of \( d^* = 6 \), such that \( S(d) \) is maximized.

In Table 3, for \( p_0 = 10^{-1}, 10^{-2}, 10^{-3}, \ldots, 10^{-7} \), we demonstrate the optimal number of levels \( |d^*| \), the corresponding optimal group sizes \( |m_1^*|, |m_2^*|, \ldots, |m_d^*| \), and the maximum speedup achieved by the \( |d^*| \)-level hierarchical pooling strategy.

In Figure 2, for \( q_0 = 0.900, 0.905, 0.910, \ldots, 0.995 \), we show the maximum achievable speedup \( S(q_0) \) of a hierarchical pooling strategy as a function of \( q_0 \). It is observed that as \( q_0 \) increases, \( S(q_0) \) increases very rapidly.
In Figure 2, we show the maximum achievable speedup $S(1/p_0)$ of a hierarchical pooling strategy as a function of the reciprocal of the fraction of infected people $1/p_0$:

$$S(1/p_0) = \frac{1/p_0}{e \ln(1/p_0)}.$$ 

It can be seen that $S(1/p_0)$ is nearly a linear function of $1/p_0$. Actually, although $S(1/p_0)$ is not really a linear function of $1/p_0$, it grows faster than any sub-linear function $(1/p_0)^{1-\epsilon}$ for any small $\epsilon > 0$. 

In Figure 3, we show the maximum achievable speedup $S(1/p_0)$ of a hierarchical pooling strategy as a function of the reciprocal of the fraction of infected people $1/p_0$. 

![Figure 2](image1.png)

**Figure 2.** Speedup vs. (1 – the fraction of infected people).

![Figure 3](image2.png)

**Figure 3.** Speedup vs. the reciprocal of the fraction of infected people.
5. Concluding remarks

We have successfully derived closed-form expressions for the optimal number of levels and the optimal group sizes of a hierarchical pooling strategy. These expressions enable us to achieve the maximum possible speedup (whose closed-form expression is also available) of a hierarchical pooling strategy. Using the results in this paper, we can quickly and easily predict the performance of an optimal hierarchical pooling strategy. For instance, if the fraction of infected people is 0.0001, an 8-level hierarchical pooling strategy can achieve speedup of nearly 400.

Disclosure statement
No potential conflict of interest was reported by the author(s).

Funding
The work of Bo Yang was supported in part by the Scientific Research Key Project of Hunan Education Department [grant number 19A077].

ORCID
K. Li http://orcid.org/0000-0001-5224-4048
Bo Yang http://orcid.org/0000-0003-3782-9093

References


[8] Saarland University, Virologists show that sample pooling can massively increase coronavirus testing capacity. Medical Xpress. 2020 Apr 29.


