ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9413685

TWO-STAGE ADAPTIVE POOLING WITH RT-QPCR FOR COVID-19 SCREENING

Anoosheh Heidarzadeh and Krishna Narayanan

Department of Electrical and Computer Engineering
Texas A&M University
College Station, TX 77843

ABSTRACT

We propose two-stage adaptive pooling schemes, 2-STAP and 2-STAMP, for detecting COVID-19 using real-time reverse transcription quantitative polymerase chain reaction (RT-qPCR) test kits. Similar to the Tapestry scheme of Ghosh *et al.*, the proposed schemes leverage soft information from the RT-qPCR process about the total viral load in the pool. This is in contrast to conventional group testing schemes where the measurements are Boolean. The proposed schemes provide higher testing throughput than the popularly used Dorfman's scheme. They also provide higher testing throughput, sensitivity and specificity than the state-of-the-art non-adaptive Tapestry scheme. The number of pipetting operations is lower than the Tapestry scheme, and is higher than that for the Dorfman's scheme. The proposed schemes can work with substantially smaller group sizes than non-adaptive schemes and are simple to describe. Monte-Carlo simulations using the statistical model in the work of Ghosh *et al.* (Tapestry) show that 10 infected people in a population of size 961 can be identified with 70.86 tests on the average with a sensitivity of 99.50% and specificity of 99.62%. This is 13.5x, 4.24x, and 1.3x the testing throughput of individual testing, Dorfman's testing, and the Tapestry scheme, respectively.

1. INTRODUCTION

There is broad consensus among epidemiologists, economists and policy makers that wide-scale testing of asymptomatic patients is the key for reopening the economy. While the benefits of testing are obvious, shortage of testing kits, reagents and the ensuing low-throughput of individual testing protocols has prevented deployment of wide-scale testing. Group testing or, pooling is an alternative way to substantially increase the testing throughput.

The idea of group testing was introduced by Dorfman [1] during World War II for testing soldiers for syphilis without having to test each soldier individually. Dorfman's scheme consists of two stages (or rounds). In the first stage, the set of people to be tested is split into disjoint pools and a test is performed on each pool. If a pool tested negative, everyone in that pool will be identified as non-infected. Otherwise, if a pool tested positive, we proceed to the second stage where all people in a positive pool will be tested individually, and then identified as infected or non-infected accordingly.

Dorfman-style testing has been implemented in the past in screening for many diseases including HIV [2], Chlamydia and Gonorrhea [3], and influenza [4]. For COVID-19, several experimental results have confirmed the feasibility of using Dorfman-style pooling [5], [6], [7], [8]. While Dorfman-style pooling is easy to implement, it is not optimal. Over the past 75 years, more sophisticated group testing schemes that provide higher testing throughput have been designed. The literature on group testing is too vast to review in detail and an overview of the techniques can be found in [9] and [10]. Group testing is also related to compressed sensing and insights from compressed sensing have been used to design group testing schemes. An important difference between the two problems is that in group testing, the measurements are Boolean (test result is either positive or negative) and they correspond to *non-linear* functions of the unknown vector.

The vast majority of the work using group testing with real-time reverse transcription quantitative polymerase chain reaction (RT-qPCR) has only considered Boolean measurements even though the RT-qPCR process can produce more fine-grained information (soft information) about the total viral load in the pool. It is well-known in information theory that such soft information can potentially be used to increase testing throughput substantially. However, group testing schemes that leverage soft information from the RT-qPCR process remain largely unexplored.

Very recently, Ghosh *et al.* in [11] developed a statistical model relating the soft information from the RT-qPCR to the total viral load in the pool. They designed a scheme called Tapestry, which uses *non-adaptive* group testing using Kirkman triples and they considered several decoding algorithms that use the soft information. They showed substantial gains in testing throughput over Dorfman's scheme and to the best of our knowledge, this scheme is the state of the art non-adaptive group testing scheme that works with RT-qPCR. A compressed-sensing based non-adaptive scheme that uses the soft information from RT-qPCR was also recently proposed in [12] for a different model than the one in [11].

This material is based upon work supported by the National Science Foundation (NSF) under Grant CCF-2027997.

Here, we propose two simple and effective *two-stage adaptive pooling* schemes that use the soft information from the RT-qPCR process and provide several advantages over Dorfman's scheme and the Tapestry scheme. Although adaptive schemes have been extensively studied for Boolean measurements, to the best of our knowledge this work is among the first to design adaptive schemes that leverage the soft information from RT-qPCR. The proposed algorithms are referred to as the Two-stage Adaptive Pooling (2-STAP) and the Two-stage Adaptive Mixed Pooling (2-STAMP) schemes/algorithms. The proposed schemes provide substantially higher throughput than Dorfman-style testing. Compared to the Tapestry scheme in [11], 2-STAP and 2-STAMP have higher testing throughput and under the statistical model developed in [11], for all tested cases, our algorithms have higher sensitivity and higher specificity. The proposed algorithms require fewer pipetting operations than Tapestry, but require more pipetting operations than Dorfman's scheme. Finally, 2-STAP and 2-STAMP work with much smaller pool sizes and population sizes than the Tapestry algorithm and hence, is easy to describe and implement in the lab. Monte-Carlo simulations using the statistical model in the work of Ghosh *et al.* [11], show that 10 infected people in a population of size 961 can be identified with 70.86 tests on the average with a sensitivity of 99.50% and specificity of 99.62% with a pool size of 31. This is 13.5x, 4.24x, and 1.3x the testing throughput of individual testing, Dorfman's testing, and the Tapestry scheme, respectively. Unlike Tapestry, which is a non-adaptive scheme, 2-STAP and 2-STAMP require storage of the swab samples and their accessibility for the second round of testing—similar to that of Dorfman's scheme.

2. PROBLEM SETUP

Consider a population of n people, labeled $1, \ldots, n$, that are to be tested for COVID-19. The vector of viral loads of these people can be modeled by a signal $\mathbf{x} = [x_1, \ldots, x_n]^\mathsf{T}$, $x_j \in \mathbb{R}_{\geq 0}$, where the jth coordinate of $\mathbf{x}$ represents the viral load of the jth person. If the jth person is infected, then x_j is a nonzero value; otherwise, if the jth person is not infected, then x_j is zero. We assume that every coordinate in $\mathbf{x}$ is nonzero with probability p (or zero with probability $1-p$), independently from other coordinates, and every nonzero coordinate takes a value from $\mathbb{R}_{>0}$ according to a fixed and known probability distribution p_x. Note that the sparsity parameter p, which is known as *prevalence* in this context, may or may not be known. We denote by $S(\mathbf{x})$ the support set of $\mathbf{x}$, i.e., the index set of all nonzero coordinates in $\mathbf{x}$.

The ith binary linear measurement y_i of $\mathbf{x}$ is defined as a linear combination of the coordinates x_j's according to the coefficients a_{ij}'s that are elements from $\{0,1\}$. That is, $y_i = \mathbf{a}_i \cdot \mathbf{x} = \sum_{j=1}^n a_{i,j} x_j$, where $\mathbf{a}_i = [a_{i,1}, \ldots, a_{i,n}]$, $a_{i,j} \in \{0,1\}$. Any coordinate x_j such that $a_{i,j} = 1$ is referred to as an *active coordinate* in the measurement y_i.

Suppose we sense the signal $\mathbf{x}$ by making the measurements $y_1, y_2, \ldots$, and observe noisy versions of $y_1, y_2, \ldots$, denoted by $z_1, z_2, \ldots$. The ith measurement y_i and the noisy measurement z_i are given by $y_i = \sum_{j=1}^n a_{i,j} x_j$ and $z_i = y_i \varepsilon_i$, respectively, where ε_i's are independent realizations of a random variable ε—taking values from $\mathbb{R}_{>0}$ according to a fixed and known probability distribution p_ε. A detailed explanation about this noise model can be found in [13].

We refer to the process of generating the measurements of $\mathbf{x}$ as *sensing*, and refer to the process of estimating the set of infected people (i.e., $S(\mathbf{x})$) from the noisy measurements as *recovery*. Given a sensing algorithm and a recovery algorithm, the average fraction of infected people that have been identified as non-infected is referred to as the *conditional false negative rate* (denoted by r_-^k), and the average fraction of non-infected people that have been identified as infected is referred to as the *conditional false positive rate* (denoted by r_+^k), where both averages are taken over all populations of size n that contain k infected people. (Here, the term "conditional" reflects that the averages are taken over all populations with a fixed number of infected people.) The quantities $1 - r_-^k$ and $1 - r_+^k$ are referred to as the *conditional sensitivity* and the *conditional specificity*, respectively.

Our goal is to design a sensing algorithm and a recovery algorithm such that the total number of measurements is minimized while the (conditional) false negative/positive rate remain below some target thresholds (e.g., below 1%), or equivalently, the (conditional) sensitivity/specificity remain above some target thresholds (e.g., above 99%).

3. 2-STAP: A TWO-STAGE ADAPTIVE POOLING

In this section, we propose a two-stage sensing algorithm and an associated recovery algorithm which we collectively refer to as the 2-STAP scheme. The first stage of 2-STAP is the same as that of Dorfman's scheme. That is, the signal coordinates are pooled into disjoint groups of equal size, and one measurement is made for each pool where all coordinates in the pool are active in the measurement. The second stage of 2-STAP, however, differs from that of Dorfman's scheme in that a number of measurements are made on not-necessarily-singleton subsets of coordinates, in each positive pool.

Given a signal $\mathbf{x}$, we partition the n signal coordinates $x_1, \ldots, x_n$ into q pools of size $s = n/q$. We denote by $\mathbf{x}_l$ the lth pool of coordinates, i.e., $\mathbf{x}_l = [x_{(l-1)s+1}, \ldots, x_{ls}]^\mathsf{T}$. We denote by m_l' and m_l'' the number of measurements for the lth pool in the first and second stage, respectively. Also, we denote by $\mathbf{A}_l'$ and $\mathbf{A}_l''$ the sensing matrix corresponding to the lth pool in the first and second stage, respectively, and denote by $\mathbf{A}_l = [(\mathbf{A}_l')^\mathsf{T}, (\mathbf{A}_l'')^\mathsf{T}]^\mathsf{T}$ the overall sensing matrix corresponding to the lth pool. $\mathbf{A}_l'$ is an $m_l' \times s$ matrix, $\mathbf{A}_l''$ is an $m_l'' \times s$ matrix, and $\mathbf{A}_l$ is an $m_l \times s$ matrix, where $m_l = m_l' + m_l''$. Let $m' = \sum_{l \in [q]} m_l'$ and $m'' = \sum_{l \in [q]} m_l''$ be the number of measurements in the first and second stage, respectively.

3.1. Sensing Algorithm for First Stage

In the following, we denote by $\mathbf{1}_t$ or $\mathbf{0}_t$ an all-one or an all-zero row vector of length t, respectively.

In the first stage, for each pool $l \in [q]$, we make one measurement $y'_l = \mathbf{1}_s \cdot \mathbf{x}_l$. That is, $m'_l = 1$ and $\mathbf{A}'_l = \mathbf{1}_s$. Thus, the total number of measurements in the first stage is $m' = q$, and the sensing matrix in the first stage, $\mathbf{A}'$, is a $q \times n$ matrix whose lth row is given by $[\mathbf{0}_{(l-1)s}, \mathbf{1}_s, \mathbf{0}_{(q-l)s}]$.

3.2. Recovery Algorithm for First Stage

Suppose we observe the noisy measurements $z'_l = y'_l \varepsilon'_l$ for $l \in [q]$. Let $L = \{l \in [q] : z'_l = 0\}$. Assume, without loss of generality, that $L = [q] \setminus [t]$ for some $0 \le t \le q$, i.e., the first t pools are positive, and the last $q - t$ pools are negative.

3.3. Sensing Algorithm for Second Stage

For any $l \in [t]$, we need to make additional measurements for the lth pool in the second stage, because such a pool contains at least one nonzero coordinate. For any $l \in [q] \setminus [t]$, the lth pool contains only zero coordinates, and we do not need to make any additional measurements for any such pool in the second stage. Intuitively, the larger is m''_l, the smaller will be the (conditional) false negative/positive rate, but the larger will be the average number of measurements. Since it is not known how to theoretically optimize m''_l, we resort to a heuristic approach to choose m''_l. We present two variants of the 2-STAP scheme: 2-STAP-I and 2-STAP-II. In 2-STAP-I, for all positive pools, the number of measurements and the sensing scheme in the second stage will be the same, regardless of the observed measurements for these pools in the first stage. In 2-STAP-II, for each positive pool, the number of measurements and the sensing scheme in the second stage will be chosen based on the number of nonzero coordinates in $\mathbf{x}_l$, denoted by k_l. Since k_l may not be known *a priori*, we compute an estimate $\widehat{k}_l$ of k_l as follows.

Let $p(k)$ be the probability that $\mathbf{x}_l$ has k nonzero coordinates and $s - k$ zero coordinates, and let $p(z'_l|k)$ be the probability density of $z'_l = y'_l \varepsilon'_l$ where $y'_l = \mathbf{1}_s \cdot \mathbf{x}_l$ given that $\mathbf{x}_l$ has k nonzero coordinates and $s - k$ zero coordinates. If the sparsity parameter p is known, for any k, $p(k)$ and $p(z'_l|k)$ can be computed exactly or approximately (depending on the distribution of values of the nonzero coordinates and the noise distribution). Given a noisy measurement z'_l, a maximum-a-posteriori (MAP) estimate of k_l is then given by $\widehat{k}_l = \mathrm{argmax}_k\, p(k)p(z'_l|k)$. If p is not known, we compute a maximum-likelihood (ML) estimate $\widehat{p} = 1 - (1 - t/q)^{1/s}$ of p, and use $\widehat{p}$, instead of p, to first compute $p(k)$ and $p(z'_l|k)$ for any k, and then compute a MAP estimate $\widehat{k}_l$ of k_l.

Given m''_l, the optimal design of $\mathbf{A}''_l$ is not known. In this work, for each $l \in [t]$, we randomly choose $\mathbf{A}''_l$ from the ensemble of all $m''_l \times s$ binary matrices (with distinct rows and columns) with a pre-specified row/column weight profile. The weight profile must be chosen to obtain a good trade-off between the computational complexity of sensing/recovery algorithms and the false negative/positive rates. The weight profiles used in our simulations can be found in [13].

3.4. Recovery Algorithm for Second Stage

For each $l \in [t]$, suppose the noisy measurement vector $\mathbf{z}''_l = \mathbf{y}''_l \boldsymbol{\varepsilon}''_l$ is observed, where $\mathbf{y}''_l = \mathbf{A}''_l \mathbf{x}_l$. Let $\mathbf{z}_l = [z'_l, (\mathbf{z}''_l)^\mathsf{T}]^\mathsf{T}$ be the overall noisy measurement vector corresponding to the lth pool. We will estimate the support set S_l of $\mathbf{x}_l$ using a recovery algorithm that consists of three steps: COMP decoding, MAP decoding, and list generation.

Combinatorial Orthogonal Matching Pursuit (COMP) Decoding: First, we use the COMP algorithm to find a (superset) estimate $\widehat{S}_l$ of S_l from $\mathbf{z}_l$ given $\mathbf{A}_l$. (A detailed explanation of the COMP algorithm can be found in [13].) Let $I_l = \{i \in [m_l] : (\mathbf{z}_l)_i = 0\}$, where $(\mathbf{z}_l)_i$ denotes the ith coordinate in $\mathbf{z}_l$. We denote by $\mathbf{x}^*_l$ the sub-vector of $\mathbf{x}_l$ restricted to the coordinates indexed by $\widehat{S}_l$; denote by $\mathbf{A}^*_l$ the sub-matrix of $\mathbf{A}_l$ restricted to the rows indexed by $[m_l] \setminus I_l$ and the columns indexed by $\widehat{S}_l$; and denote by $\mathbf{z}^*_l$ the sub-vector of $\mathbf{z}_l$ restricted to the coordinates indexed by $[m_l] \setminus I_l$. Let $m^*_l = |I_l|$ and $s^*_l = |\widehat{S}_l|$. In the next step, we will estimate the support set S^*_l of $\mathbf{x}^*_l$ from $\mathbf{z}^*_l$, given $\mathbf{A}^*_l$.

MAP Decoding: Given the estimate $\widehat{k}_l$ of k_l (the number of nonzero coordinates in $\mathbf{x}^*_l$), let $k_{\min} = \max\{\widehat{k}_l - 1, 1\}$ and $k_{\max} = \min\{\widehat{k}_l + 1, s^*_l\}$. For any $k_{\min} \le k \le k_{\max}$, and for any k-subset T of $\widehat{S}_l$, we compute

$$f(T) = \max_{\widehat{\mathbf{x}}^*_l:\ \text{support set of } \widehat{\mathbf{x}}^*_l \text{ is } T} p(\widehat{\mathbf{x}}^*_l|\mathbf{z}^*_l)$$

by finding $\widehat{\mathbf{x}}^*_l$ with support set T such that the conditional probability density of $\widehat{\mathbf{x}}^*_l$ given $\mathbf{z}^*_l$ is maximum. Maximizing $p(\widehat{\mathbf{x}}^*_l|\mathbf{z}^*_l)$ is equivalent to maximizing $p(\widehat{\mathbf{x}}^*_l)p(\mathbf{z}^*_l|\widehat{\mathbf{x}}^*_l) = \prod_{j\in[s^*_l]} p((\widehat{\mathbf{x}}^*_l)_j) \prod_{i\in[m^*_l]} p((\mathbf{z}^*_l)_i|\widehat{\mathbf{x}}^*_l)$, where $(\widehat{\mathbf{x}}^*_l)_j$ denotes the jth coordinate in $\widehat{\mathbf{x}}^*_l$. For any $\widehat{\mathbf{x}}^*_l$, $p((\widehat{\mathbf{x}}^*_l)_j) = (1 - p)\delta((\widehat{\mathbf{x}}^*_l)_j) + p \times p_x((\widehat{\mathbf{x}}^*_l)_j)$, where $\delta(x)$ is the Dirac delta function, and $p((\mathbf{z}^*_l)_i|\widehat{\mathbf{x}}^*_l) = p_\varepsilon((\mathbf{z}^*_l)_i/((\mathbf{A}^*_l)_i\widehat{\mathbf{x}}^*_l))$, where $(\mathbf{A}^*_l)_i$ denotes the ith row of $\mathbf{A}^*_l$. Thus, if the sparsity parameter p is known, $f(T)$ (for any T) can be approximated by solving a (potentially non-linear and/or non-convex) optimization problem (depending on the distributions p_x and p_ε) numerically. (In our simulations, the "fmincon" function in MATLAB was used to compute an approximation of $f(T)$.) If p is not known, $f(T)$ can be approximated similarly, except using the ML estimate $\widehat{p}$ everywhere, instead of p.

List Generation: Let $f_* = \max_T f(T)$ where the maximization is over all T defined as above. We find all T, say $T_1, T_2, \ldots, T_\ell$, such that $f(T) \ge \alpha f_*$ for a given $0 < \alpha \le 1$, and use $T_1 \cup T_2 \cup \ldots \cup T_\ell$ as the estimate of the support set S^*_l of $\mathbf{x}^*_l$. Note that the larger is the threshold α, the smaller will be the (conditional) average false negative rate and the larger will be the (conditional) average false positive rate.

Table 1. Performance results for the Tapestry scheme [11] and the proposed 2-STAP and 2-STAMP schemes

k	$m_{\min}$	$m_{\max}$	m_{std}	m_{ave}	Pooling Scheme	Conditional Sensitivity	Conditional Specificity
5	93	93	0	93	Tapestry + COMP + NN-LASSO [11]	1.00	1.00
					Tapestry + COMP + NN-OMP [11]	1.00	1.00
					Tapestry + COMP + SBL [11]	1.00	1.00
	49	61	2.94	59.08	2-STAP-I	1.00	1.00
	47	57	2.26	54.55	2-STAP-II	1.00	1.00
	46	55	2.25	52.56	2-STAMP	1.00	1.00
10	93	93	0	93	Tapestry + COMP + NN-LASSO [11]	0.98	0.99
					Tapestry + COMP + NN-OMP [11]	0.96	1.00
					Tapestry + COMP + SBL [11]	0.99	0.99
	73	91	5.54	82.96	2-STAP-I	1.00	1.00
	66	81	4.18	74.98	2-STAP-II	0.99	1.00
	63	76	3.74	70.86	2-STAMP	1.00	1.00
15	93	93	0	93	Tapestry + COMP + NN-LASSO [11]	0.94	0.97
					Tapestry + COMP + NN-OMP [11]	0.86	0.99
					Tapestry + COMP + SBL [11]	0.98	0.97
	85	121	7.31	103.30	2-STAP-I	0.98	0.99
	78	106	5.65	92.66	2-STAP-II	0.98	0.99
	74	99	5.02	86.85	2-STAMP	0.99	0.99

4. 2-STAMP: A TWO-STAGE ADAPTIVE MIXED POOLING

In this section, we propose a generalization of the 2-STAP scheme, termed the 2-STAMP scheme. The first stage of the 2-STAMP scheme is the same as that in the 2-STAP scheme, but in the second stage we make measurements on mixtures of positive pools together, instead of making measurements on separate pools only. The details of the sensing and recovery algorithms in the second stage of the 2-STAMP scheme are omitted due to the lack of space, and can be found in [13]. In the following, we briefly explain the main idea behind mixing pools in the second stage of the 2-STAMP scheme.

Consider two positive pools that we expect to contain a relatively small number of nonzero coordinates. By mixing these pools together and sensing the mixed pool altogether, we can save a few measurements while maintaining the implementation/computational complexity of the scheme affordable. The rest of the pools that are expected to contain a relatively large number of nonzero coordinates will be sensed individually, so as to avoid the scheme to become too complex implementation-wise or computationally.

5. SIMULATION RESULTS

Here, we present our simulation results. As a case study, we have considered a population of $n = 961$ people to be tested for COVID-19 and assumed that the prevalence is $p = 0.01$. For both the proposed schemes, we have considered pooling the population into $q = 31$ pools, each of size $s = n/q = 31$, in the first stage. Three different values of number of infected people in the population (k), namely $k \in \{5, 10, 15\}$, have been considered. For every k, we performed 100 Monte-Carlo simulations, where the statistical models used for viral load and measurement noise were obtained from [11].

Table 1 summarizes our results for the proposed 2-STAP (both variants) and 2-STAMP schemes and the results for the Tapestry scheme for the same problem model (i.e., the same population size and the same viral load and noise distributions). In this table, $m_{\min}$, $m_{\max}$, m_{std}, and m_{ave} represent the minimum, maximum, standard deviation, and the average of the number of measurements used in 100 simulations, respectively. The sensitivity and specificity results are rounded to two decimal places, for fair comparison with the results reported in [11]. More detailed simulation results for the proposed schemes can be found in [13].

Comparing the results of Tapestry and the two variants of 2-STAP in Table 1, it can be seen that for $k \in \{5, 10\}$, 2-STAP-I requires smaller number of measurements on the average for the same (or even higher) sensitivity and specificity. For $k = 15$, 2-STAP-I uses about 10 more measurements than Tapestry on the average, but it achieves a substantially higher specificity by about 2% for almost the same sensitivity. It can also be seen that for all $k \in \{5, 10, 15\}$, 2-STAP-II can provide higher sensitivity and higher specificity than Tapestry with even smaller (average) number of measurements. These improvements in the performance are mainly due to the fact that 2-STAP is an adaptive scheme (although with a very small degree of adaptivity, i.e., using only one round of feedback), whereas Tapestry is non-adaptive.

As can be seen in Table 1, 2-STAMP can achieve a sensitivity and a specificity higher than those attainable with 2-STAP-I and 2-STAP-II, with even smaller average number of measurements. The advantage of 2-STAMP comes from the saving in the number of measurements in the second stage. In particular, in 2-STAMP, mixing small groups of pools with small number of infected people gives rise to an opportunity for making a smaller number of measurements on the mixed super-pool without compensating the overall accuracy.

6. REFERENCES

[1] R. Dorfman, "The detection of defective members of large populations," *The Annals of Mathematical Statistics*, vol. 14, no. 4, pp. 436–440, 1943.

[2] J. Emmanuel, M. Bassett, H. Smith, and J. Jacobs, "Pooling of sera for human immunodeficiency virus (HIV) testing: an economical method for use in developing countries." *Journal of clinical pathology*, vol. 41, no. 5, pp. 582–585, 1988.

[3] J. L. Lewis, V. M. Lockary, and S. Kobic, "Cost savings and increased efficiency using a stratified specimen pooling strategy for chlamydia trachomatis and neisseria gonorrhoeae," *Sexually Transmitted Diseases*, vol. 39, no. 1, pp. 46–48, 2012.

[4] T. T. Van, J. Miller, D. M. Warshauer, E. Reisdorf, D. Jernigan, R. Humes, and P. A. Shult, "Pooling nasopharyngeal/throat swab specimens to increase testing capacity for influenza viruses by PCR," *Journal of clinical microbiology*, vol. 50, no. 3, pp. 891–896, 2012.

[5] C. R. Bilder, P. C. Iwen, B. Abdalhamid, J. M. Tebbs, and C. S. McMahan, "Tests in short supply? try group testing," *Significance (Oxford, England)*, vol. 17, no. 3, p. 15, 2020.

[6] N. Shental, S. Levy, S. Skorniakov, V. Wuvshet, Y. Shemer-Avni, A. Porgador, and T. Hertz, "Efficient high throughput SARS-CoV-2 testing to detect asymptomatic carriers," *medRxiv*, 2020.

[7] I. Yelin, N. Aharony, E. Shaer-Tamar, A. Argoetti, E. Messer, D. Berenbaum, E. Shafran, A. Kuzli, N. Gandali, T. Hashimshony, Y. Mandel-Gutfreund, M. Halberthal, Y. Geffen, M. Szwarcwort-Cohen, and R. Kishony, "Evaluation of COVID-19 RT-qPCR test in multi-sample pools," 2020.

[8] C. A. Hogan, M. K. Sahoo, and B. A. Pinsky, "Sample pooling as a strategy to detect community transmission of SARS-CoV-2," *Jama*, vol. 323, no. 19, pp. 1967–1969, 2020.

[9] D. Du and F. K. Hwang, *Combinatorial group testing and its applications*. World Scientific, 2000, vol. 12.

[10] M. Aldridge, O. Johnson, and J. Scarlett, *Group Testing: An Information Theory Perspective*. now, 2019.

[11] S. Ghosh, A. Rajwade, S. Krishna, N. Gopalkrishnan, T. E. Schaus, A. Chakravarthy, S. Varahan, V. Appu, R. Ramakrishnan, S. Ch *et al.*, "Tapestry: A single-round smart pooling technique for COVID-19 testing," *medRxiv*, 2020.

[12] W. X. Jirong Yi, Raghu Mudumbai, "Low-cost and high-throughput testing of COVID-19 viruses and antibodies via compressed sensing: System concepts and computational experiments," *arXiv preprint arXiv:2004.05759*, 2020.

[13] A. Heidarzadeh and K. Narayanan, "Two-stage adaptive pooling with RT-qPCR for COVID-19 screening," *arXiv preprint arXiv:2007.02695*, 2020.