Noisy Pooled PCR for Virus Testing

Junan Zhu, Kristina Rivera, and Dror Baron *Senior Member, IEEE*

Abstract*—Fast testing can help mitigate the coronavirus disease 2019 (COVID-19) pandemic. Despite their accuracy for single sample analysis, infectious diseases diagnostic tools, like RT-PCR, require substantial resources to test large populations. We develop a scalable approach for determining the viral status of pooled patient samples. Our approach converts group testing to a linear inverse problem, where false positives and negatives are interpreted as generated by a noisy communication channel, and a message passing algorithm estimates the illness status of patients. Numerical results reveal that our approach estimates patient illness using fewer pooled measurements than existing noisy group testing algorithms. Our approach can easily be extended to various applications, including where false negatives must be minimized. Finally, in a Utopian world we would have collaborated with RT-PCR experts; it is difficult to form such connections during a pandemic. *We welcome new collaborators to reach out and help improve this work!

***Index Terms*—Approximate message passing, COVID-19, group testing, linear inverse problems, pooling, RT-PCR, virus.**

I. Introduction

Motivation. *Reverse transcription polymerase chain reaction* (RT-PCR) is a prevalent diagnostic tool for infectious diseases, like *coronavirus disease 2019* (COVID-19). RT-PCR is a labor intensive technical procedure that requires numerous trained laboratory personnel to analyze one patient sample [1]. Briefly, *ribonucleic acid* (RNA) is isolated from a patient's respiratory tract and purified for reverse transcription, a process where the RNA template is turned into *complementary deoxyribonucleic acid* (cDNA). cDNA, along with specific viral primers, is loaded into a machine, where cDNA is amplified and annealed to the target sequence. While extending through each PCR cycle, a reporter dye is cleaved or broken from a probe to amplify fluorescence intensity and reveal a positive sample.

Despite its accuracy for single sample analysis, RT-PCR requires substantial resources to test a large number of samples. Instead, we aim to develop a scalable testing procedure that allows for patient samples to be combined before PCR.

Main idea. Noisy group testing is used to analyze RT-PCR data from mixed or *pooled* samples, as recently demonstrated for COVID-19 [2]. The goal of group testing is twofold. First, to increase the accuracy of testing for each individual patient by combining information from multiple pooled measurements that sample genetic material from that same individual. Second, to use a reduced number of measurements, especially in settings where a large population is being tested, most patients are healthy, and so many individual measurements will come out negative and thus show that multiple patients are healthy. In summary, group testing allows to evaluate large populations at high throughput, low per-patient diagnostic costs, and low false positive and negative probabilities.

Noiseless group testing has been established, but noisy group testing algorithms are less mature. For example, recent work on COVID-19 [2], [3] uses pooled tests to rule out patients corresponding to negative pooled measurements. Their approach implicitly relies on false negatives being rare in RT-PCR, but diluting many samples may increase false negatives [4]. Additionally, patients corresponding to positive pooled measurements are later tested individually [3], which does not benefit from pooling. Our algorithm (Sec. III) applies pooling to identify individual sick patients.

Recently, researchers across the world have been looking to increase the sample size per PCR run by using custom barcodes for each sample and then pooling them together [5]. Custom barcoding is not new in terms of multiplexed genetic sequencing [6]. Briefly, custom barcodes for each patients RNA sample are designed by an algorithm and substituted in as the *reverse transcriptase* (RT) primers to generate barcoded cDNA. Next-generation sequencing is performed after a single pooled PCR reaction, and then demultiplexed to determine each samples viral content. Our method of pooling samples before adding barcodes could be used by researchers for a quicker time to analysis and also as a complementary method to reanalyze barcoded samples.

Contributions and organization. We focus on a simple pooled testing model for RT-PCR (Sec. II). This model is converted to a linear inverse problem (Sec. II-A), and our goal is to estimate a vector of patient illness status, x, from a vector of noisy RT-PCR measurements, y, a matrix A relating patients and measurements, and statistical information about false positives and negatives (Sec. II-B and Fig. 1). This estimation problem is solved using *generalized approximate message passing* (GAMP) [7] in Sec. III. Promising numerical results are provided in Sec. IV, and Sec. V discusses how our GAMP-based approach can be extended.

II. Model

Conventional RT-PCR. RT-PCR has a binary outcome. That is, once the sample is amplified, there is plenty of genetic material available for identification. Prior to amplification, there can be problems during pre-processing to isolate purified RNA. Either genetic material can be damaged, in which case all further tests with this material are negative, or the sample is contaminated, in which case further tests are positive. However, such pre-processing problems essentially flip the sample's condition permanently from negative to positive or

The work of Baron was supported in part by NSF EECS #1611112. Rivera was supported by the National Institutes of Health under award number F31DK118859-02 and by the 2019 Howard Hughes Medical Institute Gilliam Fellowship Award.

Zhu is with Harvest Fund Management, Beijing, China, email junan.zhu@gmail.com. Rivera and Baron are with North Carolina State University, Raleigh, NC, email {krrivera, barondror}@ncsu.edu.

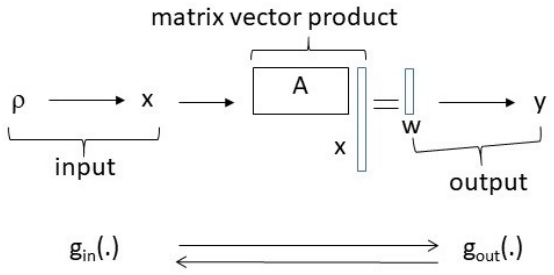


Fig. 1. System model. A Bernoulli process with probability ρ of sickness generates an input vector $x \in \{0,1\}^N$, which reflects patient illness status. The input is multiplied by a measurement matrix, $A \in \{0,1\}^{M\times N}$, resulting in noiseless measurements, $w = Ax \in \mathbb{N}^M$ (1), which are processed by an RT-PCR channel, resulting in noisy measurements, $y \in \{0,1\}^M$ (2). GAMP [7] processes the input channel relating ρ and x with $g_{in}(\cdot)$ (4), and the output channel relating w and x with $g_{out}(\cdot)$ (5) (details in Sec. III).

vice versa, because all tests of the patient will be using flawed genetic material from this patient. Therefore, such problems will not be discussed further.

Once the sample has been pre-processed, there are two further problems that could arise once we partition the sample into multiple group test measurements. One possible outcome is a *false negative*, meaning not enough genetic material from a sick patient and, therefore, insufficient amplification. It is also possible to have a *false positive*, meaning the sample was contaminated by viral matter, and the test is positive although the patient is healthy. We focus on these two problems, as a well-designed group testing procedure mitigates their effects.

Group testing. Instead of sampling genetic material from one patient, we will pool material from multiple patients. In principle, if any of the patients is sick, and there is sufficient genetic material, then the pooled group test will come out positive. However, it is possible that group testing will use less genetic material per patient, meaning that the measurement is diluted, and the probability of false negatives (per sick patient) might be larger than conventional (unpooled) measurements [4].

A. Number of sick patients per measurement

We now express the number of sick patients pooled per measurement as a product between a binary matrix representing what patients are pooled in different measurements, and a binary vector representing patient illness status. Later, Sec. II-B forms a noisy probabilistic outcome, where the RT-PCR test being positive or negative depends probabilistically on the number of sick patients pooled per measurement.

Our system is illustrated in Fig. 1. We have N patients. The status of each patient is given by x_n, where $n \in \{1,\dots,N\}$. If patient n is sick, then $x_n = 1$, else $x_n = 0$. The N entries are modeled as *independent and identically distributed* (i.i.d.), and we model x_n using a *random variable* (RV), X_n. These N RVs follow a Bernoulli *probability mass function* (pmf), $X_n \sim \text{Ber}(\rho)$,

$$\Pr(X_n = 1) = \rho \quad \text{and} \quad \Pr(X_n = 0) = 1-\rho,$$

where $\Pr(\cdot)$ denotes probability, and ρ is the percentage of sick patients.

Next, the vector $x \in \{0,1\}^N$ is multiplied by a binary *measurement matrix* $A \in \{0,1\}^{M\times N}$. Because x and A are binary, the matrix vector product,

$$w = Ax \in \mathbb{N}^M, \tag{1}$$

is a length-M vector of natural (non-negative) numbers.

The matrix A is interpreted as follows. Row m corresponds to measurement m, and column n to patient n, where $m \in \{1,\dots,M\}$ and $n \in \{1,\dots,N\}$. If patient n is not measured in measurement m, then A_{mn}, the matrix entry in row m and column n, is zero; such patients do not affect the outcome of measurement m. In contrast, $A_{mn} = 1$ when genetic material from patient n appears in measurement m. It can be seen that w_m counts the number of sick patients evaluated by measurement m.

In noiseless group testing, the RT-PCR measurement is positive if and only if $w_m > 0$. However, RT-PCR suffers from false positives and negatives.

B. Noisy model

We now account for these false positives and negatives. The noisy measurement y_m depends on w_m through a conditional probability, $\Pr(Y_m|W_m)$, where Y_m and W_m are RVs. To evaluate $\Pr(Y_m|W_m)$, we denote the probability of an individual patient being sick yet not having enough genetic material in one of the measurements by p_1. This is the probability of a false negative caused by *one* patient; with w_m sick patients, the probability that all of them have false negatives is $(p_1)^{w_m}$. (We note in passing that false negatives could be modeled as independent of w_m [3].)

Similarly, the probability of a false positive is denoted by p_2. If there is a false positive in y_m, then $y_m = 1$, irrespective of the status of the patients being evaluated by measurement m. On the other hand, $y_m = 0$ means that there was no false positive, and all the patients evaluated that were actually sick resulted in false negatives. Based on this discussion, we can express the probability for y_m to be 0 or 1 given w_m,

$$\Pr(Y_m = 0|W_m = w_m) = (1-p_2)(p_1)^{w_m}. \tag{2}$$

Then, we compute $\Pr(Y_m = 1|W_m) = 1 - \Pr(Y_m = 0|W_m)$. In communication and information theory, such a probabilistic relationship is known as a *channel* [8]; the output channel relating the vectors w and y appears in Fig. 1.

We now have a linear relationship from x to w, and the noiseless measurements vector w, which contains the number of sick patients per measurement, is then processed by a probabilistic channel to yield the noisy measurements vector, y. Our goal is to estimate x from y, A, and statistical information about the channel. Other group testing approaches often perform pooled measurements in a first part, and positives are tested individually in a second part [3]; our method can improve both parts by pooling all measurements and accounting for all available information. In the following section, we describe our algorithmic framework in detail.

Algorithm 1 GAMP

Inputs. Maximum iterations t_{max}, percentage of sick patients ρ, false negative probability p_1, false positive probability p_2, measurements y, and matrix A.

Initialize. $t, k, h_m, \Theta_m, \widehat{x}_n, s_n, \forall m, n$.

Comment. t is iteration number, k is mean of our estimate for Ax, h_m is correction term for w_m, Θ_m is variance of h, $\widehat{x}_n$ is our estimate for x_n, s_n is variance in our estimate $\widehat{x}_n$.

1: **while** $t < t_{max}$ **do**
2: // **clean up output channel**
3: $\Theta = (A)^2 s$ // variance of h
4: $k = A\widehat{x} - \Theta h$ // mean of w per previous iteration
5: **for** $m = 1$ to M **do**
6: $h_m = g_{out}(k_m, y_m, \theta_m)$
7: **Comment:** $\frac{1}{\Theta}(E[W_m|K_m, Y_m, \Theta_m] - k_m)$.
8: $r_m = -\frac{\partial}{\partial k_m} g_{out}(\cdot)$
9: $\Delta_v = \left\{\frac{1}{N}(A^T)^2 r\right\}^{-1}$ // scalar channel noise variance
10: $q = \widehat{x} + \Delta_v A^T h$ // pseudo data
11: // **clean up input channel**
12: **for** $n = 1$ to N **do**
13: $\widehat{x}_n = g_{in}(\Delta_{vn}, q_n) = E[x_n|q_n]$ // mean estimate
14: $s_n = E[x_n^2|q_n] - E^2[x_n|q_n]$ // variance estimate
15: $t = t + 1$

Output. Estimate $\widehat{x}$, pseudo data q, and scalar channel noise variance Δ_v.

III. Algorithmic Framework

We estimate x from y, A, and statistical information about the channel by applying *generalized approximate messsage passing* (GAMP) [7], which is an iterative signal estimation algorithm. GAMP is preferred, because it achieves best-possible estimation-theoretic performance asymptotically, in the limit of large linear estimation problems.

Our approach focuses on the *large system limit*, where $N \to \infty$, $M(N)$ depends on N, and $\lim_{N\to\infty} \frac{M(N)}{N} = R$, where we call R the measurement rate. GAMP relies on the large system limit for various summations in the derivation steps of the algorithm to be well-approximated as Gaussian under the central limit theorem [7]. Note that running our algorithm for small problem sizes such as $N = 100$ patients and $M = 30$ measurements may result in poor estimation quality.

The GAMP algorithm is listed in Algorithm 1. For a detailed derivation, we refer the reader to Rangan [7]. An intuitive and less formal explanation is provided below.

GAMP is comprised of two parts. The first part involves the input channel relating ρ and x (Fig. 1) [7], where x is estimated from an auxiliary vector $q \in \mathbb{R}^N$ (cf. Line 10 of Algorithm 1) through a function $g_{in}(\cdot)$. The auxiliary vector, q, is known in the AMP literature[1] as the pseudo data, which can be treated as a noisy version of the true signal x,

$$q = x + v, \tag{3}$$

where $v \in \mathbb{R}^N$ is *additive white Gaussian noise* (AWGN) with zero mean, where Δ_{vn} is the variance of v_n. Hence, $g_{in}(\cdot)$ can be interpreted as a denoising function,

$$\widehat{x}_n = g_{in}(\Delta_{vn}, q) = E[X_n|Q_n = X_n + \mathcal{N}(0, \Delta_{vn})]. \tag{4}$$

While other denoising functions can be used, conditional expectation, i.e., $E[X|Q]$, minimizes the *mean squared error* (MSE) in each GAMP iteration, and so it reduces the error as quickly as possible.

The second part of GAMP involves the output channel (cf. Fig. 1), where y_m depends probabilistically on w_m. We estimate w_m from y_m using a second denoising function,

$$h_m = g_{out}(k_m, y_m, \Theta_m) = \frac{E[W_m|K_m, Y_m, \Theta_m] - k_m}{\Theta}, \tag{5}$$

where the expectation is taken over the pmf,

$$f(w_m|k_m, y_m, \Theta_m) \propto \Pr(y_m|w_m) \exp\left[-\frac{(w_m - k_m)^2}{2\Theta_m}\right].$$

In this expression, (5), we have mean and variance values for w_m, and can interpret $g_{out}(k_m, y_m, \Theta_m)$ as a correction term that reflects residual information, which is provided by the noisy measurements vector, y, but is not yet reflected in our estimates, k for Ax, and $\widehat{x}$. The correction term is used in later iterations to compute q and $g_{in}(\cdot)$.

GAMP uses these two scalar functions, $g_{in}(\cdot)$ (4) and $g_{out}(\cdot)$ (5), to estimate x and $w = Ax$ (1) from q (3) and y (2), respectively. That is, GAMP iteratively cleans the input and output channels. A numerical illustration is provided in Fig. 2; Sec. IV discusses this figure in detail. GAMP also uses derivatives of these scalar functions to estimate the variance. In words, knowing not only the mean but also the variance around the mean allows GAMP to judiciously use information from $\widehat{x}$ when estimating $\widehat{w}$ and vice versa.

IV. Numerical Results

A. GAMP illustration

This section provides numerical results showing how GAMP solves noisy group testing problem. For readers who are new to GAMP, we begin by illustrating how GAMP cleans up the input and output channels iteratively.

We evaluate $N = 5000$ patients at a time, where the fraction of infected patients is $\rho = 0.01$. The measurement rate is $R = M/N = 0.3$, meaning that we take $M = NR = 1500$ RT-PCR measurements.[2] The matrix A is designed to pick up n_{pos} sick patients per measurement on average; we let $n_{pos} = 0.5$. The numbers of ones per row and column are kept close to n_{pos}/ρ and Rn_{pos}/ρ, respectively. For the RT-PCR channel, we assume a false negative probability, $p_1 = 0.02$, and false positive probability, $p_2 = 0.001$.[3] We quantify GAMP signal estimation quality using the *area under the receiver operating curve* (AUC-ROC). In words, the ROC captures trade-offs between false positives and negatives, and

[1] AMP can be derived from GAMP for a specific setting [7]. While AMP [9] requires the matrices it processes to have zero mean, GAMP is less restrictive.

[2] Our GAMP-based algorithm is relatively fast; problems of size ($M = 1500, N = 5000$) take a few seconds to run on a laptop computer.

[3] The parameters p_1 and p_2 resemble Hanel and Thurner [3]; other sources suggest larger false positive and negative probabilities. For our software, these are merely parameters that are easily modified.

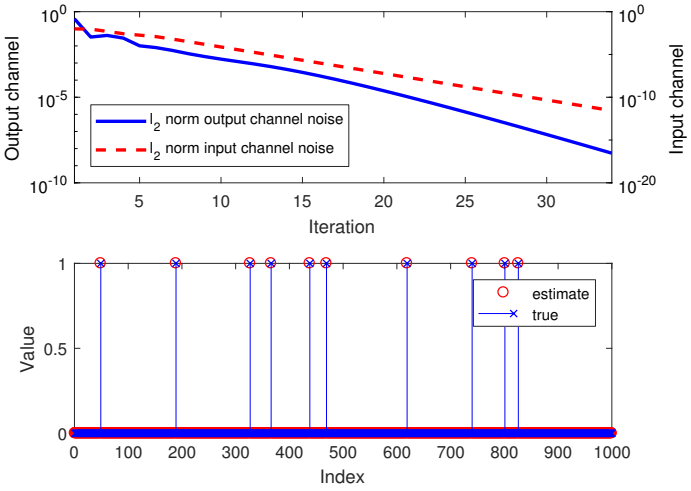


Fig. 2. Top: ℓ_2 norms of the input (dashed red line; associated with right vertical axis) and output (solid blue; left) channel noise as functions of the GAMP iteration. Bottom: The first 1000 entries of the unknown patient illness status vector x, and their estimates. ($N = 5000$ patients; $\rho = 0.01$ percentage of sick patients; measurement rate $R = M/N = 0.3$; $n_{pos} = 0.5$ average sick patients per measurement; false negative probability $p_1 = 0.02$; false positive $p_2 = 0.001$.)

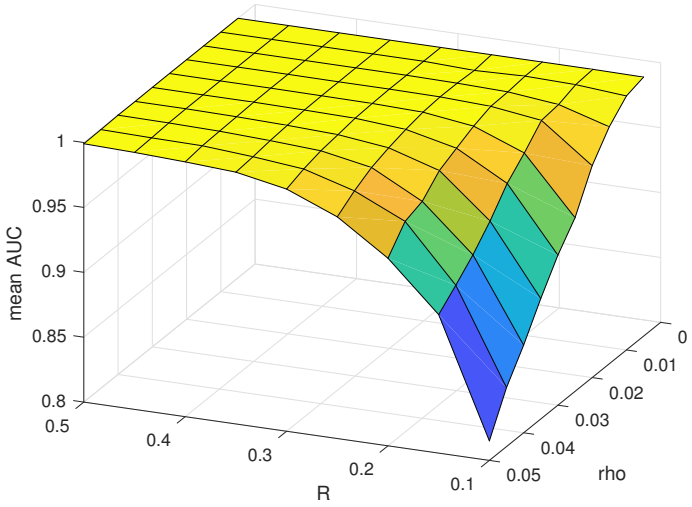


Fig. 3. Estimation accuracy in AUC (vertical axis) as a function of the percentage of sick patients, ρ, and measurement rate, $R = M/N$. ($N = 5000$; $n_{pos} = 0.5$; $p_1 = 0.02$; $p_2 = 0.001$.)

increasing the AUC reflects better estimation. While standard GAMP minimizes the MSE [7], other error metrics can be minimized [10], [11].

The top pannel of Fig. 2 plots the ℓ_2 norms of the input channel noise (dashed red line) and output channel noise (solid blue). We can see that the input and output channels improve over iterations. The bottom panel shows the first 1000 entries of the input signal vector x and their estimates. We can see that patient illness status is estimated well.

B. Group testing under various conditions

We investigate the impact of the percentage of sick patients, ρ, and measurement rate, $R = M/N$, on estimation accuracy in Fig. 3. As before, $N = 5000$, $n_{pos} = 0.5$, $p_1 = 0.02$, and $p_2 = 0.001$. We run our algorithm on $\rho \in \{0.005, 0.01, 0.015, \cdots, 0.05\}$ and $R \in \{0.1, 0.15, 0.2, \cdots, 0.5\}$. For each setting, we randomly generate 20 different triples of (x, A, y), and record the AUC for every triple. The performance for each setting is evaluated by averaging AUC values. Our results show that the AUC increases with the measurement rate, R, and larger ρ requires larger R to yield an AUC near 1. These results align with our expectation that more measurements improve estimation, while more sick patients require more measurements.

C. Two part approach

Recent work Hanel and Thurner [3] analyzes a two part group testing approach. Their model for PCR uses $\Pr(Y_m = 0|W_m > 0) = (1 - p_2)p_1$, while we use $\Pr(Y_m = 0|W_m = w_m > 0) = (1 - p_2)p_1^{w_m}$; we evaluated their approach using $\rho = 0.01$, $p_1 = 0.02$, and $p_2 = 0.001$. Hanel and Thurner's Part 1 pools a block of $B = 11$ patients at a time. If a pool is negative, all patients in the block are declared healthy; else Part 2 measures them individually. The measurement rate is

$$R = \frac{1}{B} + \Pr(\text{pool tests positive}) = \frac{1}{B} + \Pr(Y_m = 1).$$

We compute $\Pr(Y_m = 1)$,

$$\begin{aligned}\Pr(Y_m = 1) &= \Pr(W_m = 0)\Pr(Y_m = 1|W_m = 0) \\ &+ \Pr(W_m > 0)\Pr(Y_m = 1|W_m > 0).\end{aligned}$$

Note that $\Pr(W_m = 0) = (1 - \rho)^B$, $\Pr(Y_m = 1|W_m = 0) = p_2$, $\Pr(W_m > 0) = 1 - \Pr(W_m = 0)$, and

$$\Pr(Y_m = 1|W_m > 0) = 1 - (1 - p_2)p_1.$$

Combining these results,

$$R = \frac{1}{11} + 0.99^{11}0.001 + (1 - 0.99^{11})(1 - 0.999 \cdot 0.02) = 0.1944.$$

A simulation over $N = 10^7$ patients had 935 false positives and 4038 false negatives.

Our two part approach modifies Part 2. Instead of testing patients within each positive block individually, we combine all patients within all positive blocks into a new linear inverse problem, and solve the resulting estimation problem (1) with GAMP. For example, let the block size be $B = 25$ patients in Part 1. In Part 2, we combine all positive blocks and apply $R = 0.5$ and $n_{pos} = 0.5$ to the linear inverse problem. Note that (*i*) positive measurements from Part 1 are reused in the matrix A and measurement vector y of Part 2, because they contain information that helps GAMP; (*ii*) we decide whether a patient is sick or not by thresholding $\widehat{x}$. Combining Parts 1 and 2, the measurement rate is $R = 0.149$. We randomly generate 100 (x, y, A) triples for $N = 10000$ patients in Part 1, Among $100N = 10^6$ patients, there are 92 false positives and 366 false negatives. Our false positive and negative rates are both lower than those of Hanel and Thurner [3].

V. Discussion

Our current approach relies on various assumptions. Below are issues that can be considered in ongoing and future work.

Challenges. Some of the challenges we expect involve better modeling of RT-PCR, in particular how pooling multiple samples dilutes the genetic material and may increase false positives and negatives [12]. Other challenges involve matrix design; better matrices will improve estimation quality.

One question is whether p_1 is the same for each patient n in each measurement m where $A_{mn} = 1$. It might be possible to use different matrix entry values (not just 0 and 1) and thus sample more genetic material in some cases, and less in others. This will likely result in p_1 depending on the amount of genetic matter being sampled. Therefore, genetic material will have to be measured before samples are pooled to ensure the same concentration of genetic material is loaded for each sample. Limiting the amount of genetic matter being processed may reduce the costs of the overall measurement process. On the other hand, if the amount of genetic material per patient per measurement varies, a sophisticated channel could be supported by having nonzeros in A_{mn} take different values.

Another question is whether the false positive probability, p_2, is identical for all measurements. Alternately, p_2 may depend on the measurement system, for example the number of samples pooled together, or the number of RT-PCR iterations. If individual RT-PCR iterations are costly, then the number of iterations can be reduced, resulting in larger p_2. Our experience with AMP-based algorithms suggests that a modest increase in $R = M/N$ will compensate for the degraded individual measurements. Cost effectiveness of each iteration will determine the number of times each sample can be run. This value should be weighed against the number of times individual samples are pooled, allowing to optimize the number of samples pooled per measurement à la [3].

A third challenge pertains to the measurement matrix, A. In our current design, rows and columns contain similar numbers of nonzeros (Sec. IV). Therefore, each measurement provides the same *signal to noise ratio* (SNR). One matrix design option is to allow different rows and columns to have different numbers of nonzeros. Another is to prevent any pair of patients, n_1 and n_2, from both having nonzero matrix entries in different rows, m_1 and m_2, i.e., $A_{m_1n_1}$, $A_{m_1n_2}$, $A_{m_2n_1}$, and $A_{m_2n_2}$ cannot all be nonzero. Refinements in matrix design will improve our estimation quality.

Applications. Improvements in testing accuracy can be used in different ways in different applications.

•False positive and negative rates of individual RT-PCR measurements can be reduced by pooling together samples, and using GAMP-based algorithms. This can help decide when it is safe to release a COVID-19 patient from quarantine.

•Latency reduction. As the first RT-PCR measurements from a batch of patients arrive, all patients corresponding to positive measurements can be quarantined. As more RT-PCR measurements come in, GAMP can determine which of the individual patients in the positive pooled samples are actually healthy.

•Throughput can be drastically increased for fixed target levels of false positives and negatives. This can be useful for testing large populations with minimal cost.

•False negatives must be low to prevent a few sick patients from infecting many others. Low false negative probabilities can be provided by a two-part signal estimation approach [13]. Part 1 will use a conventional measurement matrix A. Part 2 takes extra measurements only for patients deemed healthy in Part 1, thus reducing false negatives. Similar ideas have been proposed for adaptive sensing [14].

Finally, our GAMP-based approach uses statistical information about the probability of a patient being sick, ρ, and probabilities governing false positives and negatives, p_1 and p_2. The algorithm can be improved using more information, for example statistical dependencies between household members being sick. We will integrate more predictive information to further improve estimation quality.

VI. Acknowledgments

The authors thank Michael Daniele for RT-PCR modeling discussions, and Ahmad Beirami, Yitzhak (Tsahi) Birk, Igor Carron, Steven Cotten, Florent Krzakala, John Muth, and Lenka Zdeborova for discussions of group testing. Baron also thanks numerous colleagues at North Carolina State University for putting him in touch with various specialists.

References

[1] T. Nolan, R. Hands, and S. Bustin, "Quantification of mRNA using real-time rt-PCR," *Nature Prot.*, vol. 1, pp. 1559–82, Feb 2006.

[2] 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," *medRxiv*, 2020. [Online]. Available: https://www.medrxiv.org/content/early/2020/03/27/2020.03.26.20039438

[3] R. Hanel and S. Thurner, "Boosting test-efficiency by pooled testing strategies for SARS-CoV-2," *arXiv preprint arXiv:2003.09944*, Mar 2020.

[4] S. L. Stramer, D. E. Krysztof, J. P. Brodsky, T. A. Fickett, B. Reynolds, R. Y. Dodd, and S. H. Kleinman, "Comparative analysis of triplex nucleic acid test assays in United States blood donors," *Transfusion*, vol. 53, no. 10pt2, pp. 2525–2537, 2013.

[5] S. R. Ayaan Hossain, Alexander C. Reis and H. M. Salis, "A Massively Parallel COVID-19 Diagnostic Assay for Simultaneous Testing of 19200 Patient Samples," *Google Docs*, Mar 2020.

[6] M. Baym, S. Kryazhimskiy, T. D. Lieberman, H. Chung, M. M. Desai, and R. Kishony, "Inexpensive multiplexed library preparation for megabase-sized genomes," *PloS one*, vol. 10, no. 5, p. e0128036, 2015.

[7] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inf. Theory*, 2011, pp. 2168–2172.

[8] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley-Interscience, 2006.

[9] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Academy Sci.*, vol. 106, no. 45, pp. 18 914–18 919, Nov. 2009.

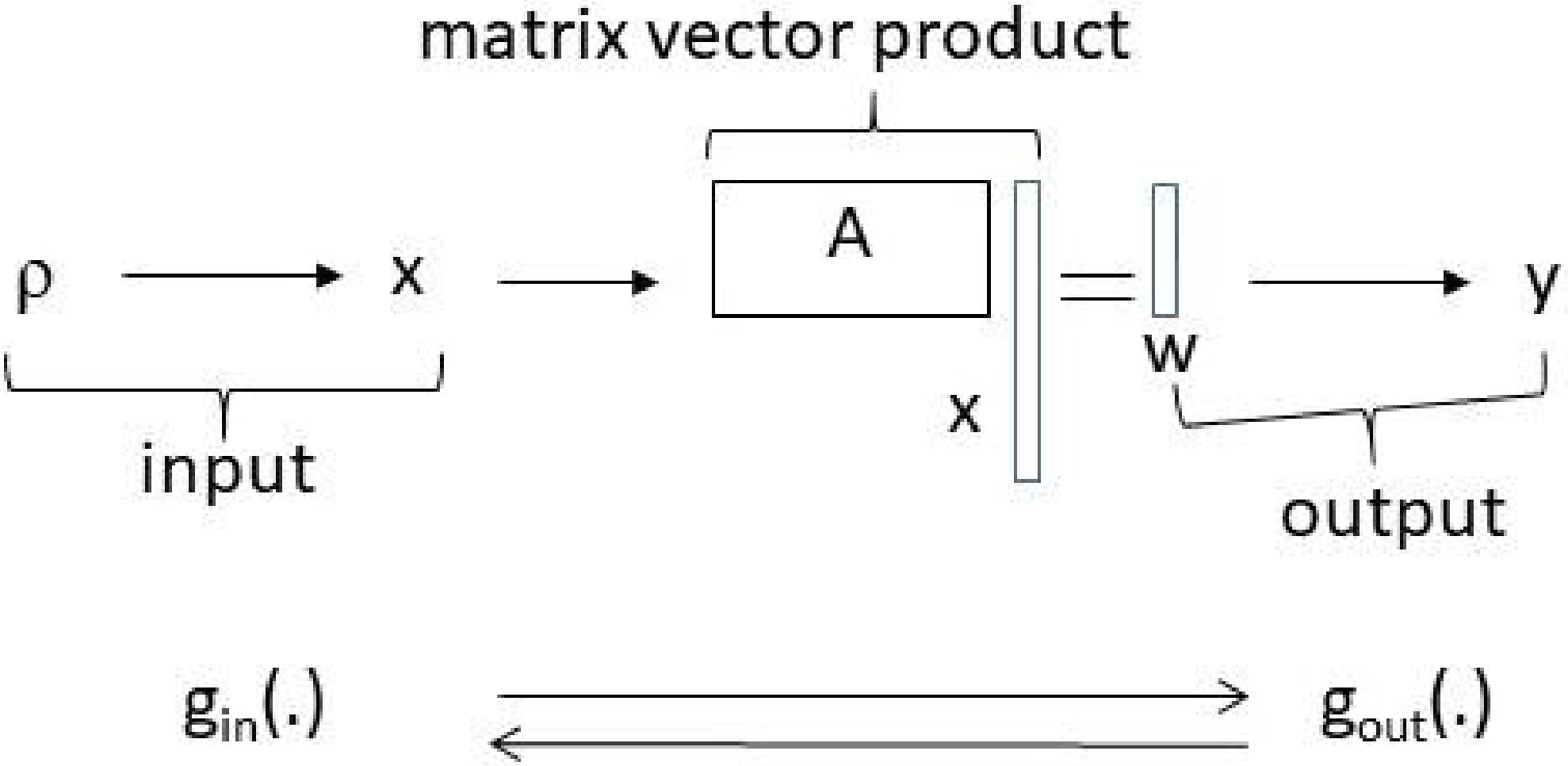
[10] J. Tan, D. Carmon, and D. Baron, "Signal estimation with additive error metrics in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 60, no. 1, pp. 150–158, Jan. 2014.

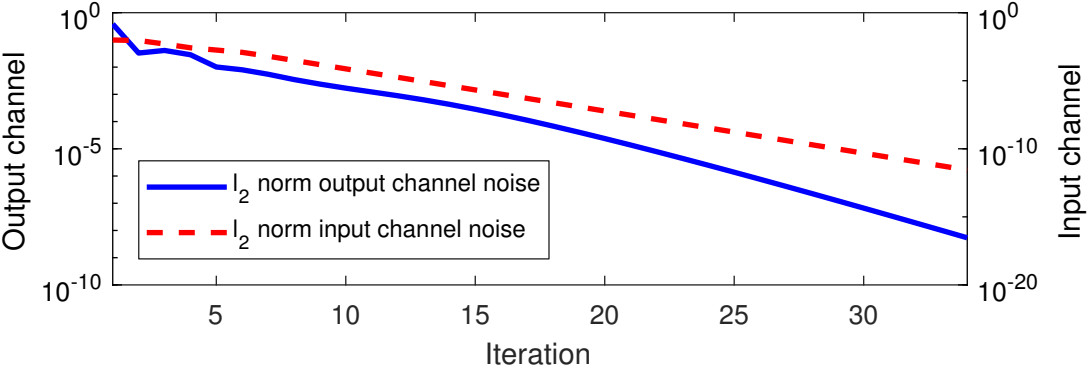
[11] J. Zhu and D. Baron, "Performance limits with additive error metrics in noisy multimeasurement vector problems," *IEEE Trans. Signal Proc.*, vol. 66, no. 20, pp. 5338–5348, Oct 2018.

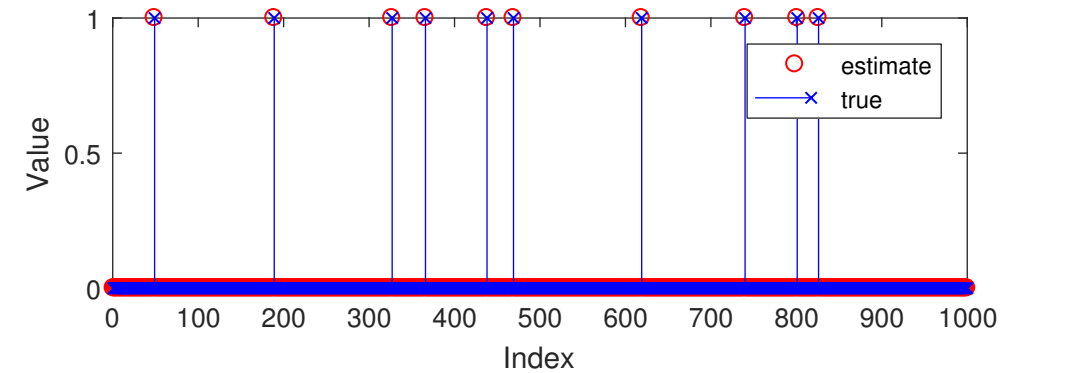
[12] L. Châtel, X. Yang, F. Cholette, H. Soudeyns, P. Sandstrom, and C. Lavigne, "Impact of pre-amplification conditions on sensitivity of the tat/rev induced limiting dilution assay," *Archives of virology*, vol. 163, no. 10, pp. 2701–2710, 2018.

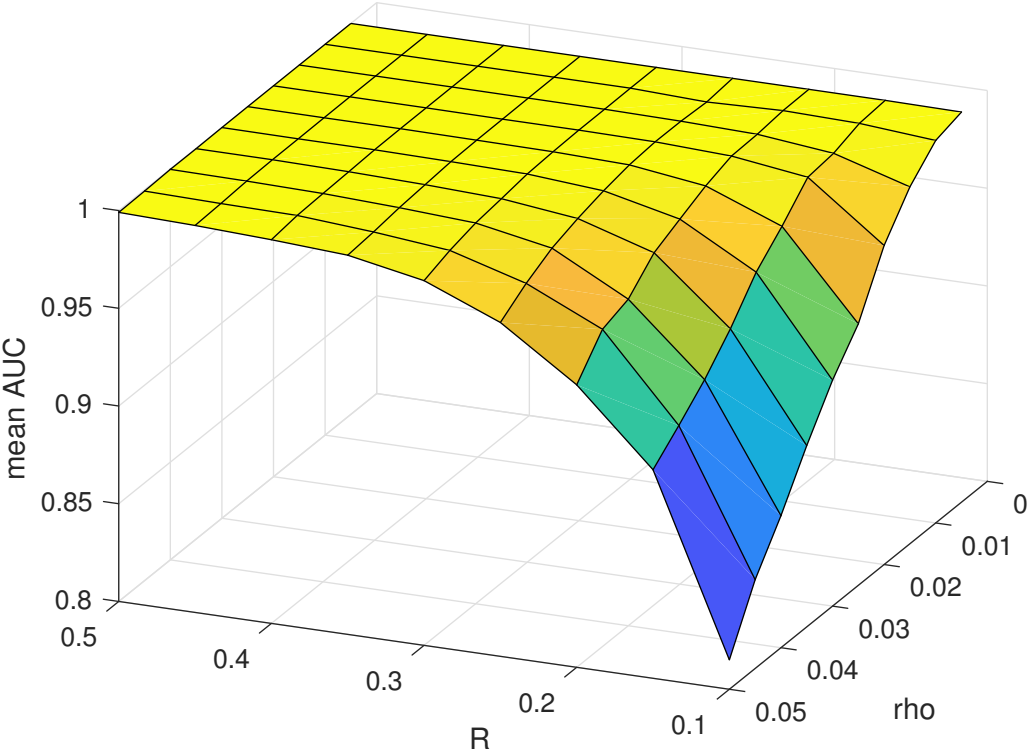
[13] Y. Ma, D. Baron, and D. Needell, "Two-part reconstruction with noisy-sudocodes," *IEEE Trans. Signal Process.*, vol. 62, no. 23, pp. 6323–6334, Dec. 2014.

[14] J. D. Haupt and R. Nowak, "Adaptive sensing for sparse recovery," in *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012.


matrix vector product
A
ρ
x
y
w
x
input
output
$g_{in}(.)$
$g_{out}(.)$


Output channel
Input channel
Iteration
l_2 norm output channel noise
l_2 norm input channel noise
10^0
10^{-5}
10^{-10}
10^{-10}
10^{-20}
5
10
15
20
25
30


Value
Index
estimate
true
1
0.5
0
0
100
200
300
400
500
600
700
800
900
1000


mean AUC
1
0.95
0.9
0.85
0.8
0.5
0.4
0.3
0.2
0.1
R
0
0.01
0.02
0.03
0.04
0.05
rho