# Efficient designs for two-round group testing at a wide range of positivity rates

Benoît Kloeckner, Pierrick Tolle

# *Efficient designs for two-round group testing at a wide range of positivity rates*

***Efficient designs for group testing***

Benoît R. Kloeckner [*] Pierrick Tolle [*†]

November 23, 2022

[*]Univ Paris Est Creteil, Univ Gustave Eiffel, CNRS, LAMA, F-94010 Creteil, France
benoit.kloeckner@u-pec.fr
[†]Ecole Normale Superieure de Lyon, France
pierrick.tolle@ens-lyon.fr

**Abstract**

As waves of Coronavirus disease (Covid-19) keep spreading around the world, increasing the throughput of testing facilities stays a matter of concerns, propelling the interest in group testing. However the matter of finding optimal group testing designs is difficult; we use *in silico* experiment to determine efficient designs for two-rounds group testing, under the practical constraints presented in the literature for single reverse transcriptase-polymerase chain reaction testing of severe acute respiratory syndrome coronavirus 2. The designs we found out to be most effective improve throughout the spectrum of positivity rate $p$ on all designs we could find in the literature for this task, and our results can be used to fine-tune the pooling strategy with the positivity rate. We propose a new family of designs, *duals of complete graphs*, which performs very well at both medium-high (7%–13%) and low ($\leq .4\%$) positivity rate, obtaining a twenty-fold reduction in the needed number of tests at $p < .1\%$. We also give a mathematical argument indicating that at small positivity rate, improving on dual of complete graphs would be likely to imply the use of impractically large designs.

**Keywords:** group testing, SARS-CoV-2, hypergraphs.

**Statements relating to ethics and integrity policies:** all source codes and data generated will be made available in a open repository. The authors acknowledge the financial support of the internship of P. Tolle during the first part of this project from the *Laboratoire d'Analyse et Mathématiques Appliquées* UMR8050, Univ Paris Est Creteil, Univ Gustave Eiffel, CNRS. The authors declare no conflict of interest.

# 1 Introduction

Testing and rational optimization of resources are some of the key elements fostering an efficient response to the Covid-19 crisis [HDFA+21]. An important idea to optimize testing is to test pools regrouping aliquots of several individual samples [Dor43]; it has recently been widely studied for application to severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [BMR21, GRK+20, LTM+21, MSLH+21, MNB+20, SLW+20, SAKH20, VP20].

In the present article, we are interested in the following question: how to design the pools in the most efficient way, given the current positivity rate? A simple computation, provided e.g. in [ACFSL20], enables to optimize the pool size when performing Dorfman's original single pool strategy. However, strategies involving overlapping pools can be much more efficient, leaving us with the difficult task of optimizing the arrangement of pools.

In this article, we present *in silico* experiments comparing several hundreds of different designs through a large spectrum of positivity rates. We restrict to designs that respect the constraints that have been identified by previous works to make them usable *in practice* for single reverse transcriptase-polymerase chain reaction (RT-PCR) testing of SARS-CoV-2. Previous articles focused on one design or on restricted families, and the main contribution of the present work is to provide a more systematic comparison, drawing both from the designs previously used for group testing for SARS-CoV-2 and from the wealth of combinatorical structures build by mathematicians. A new class of designs is introduced, *duals of complete graphs*, which bests all others in two different regimes of the positivity rate ($\leq .4\%$ and 7%–13%).

This article is empirical in nature, and one could ask *why* some designs are more efficient than others. In the Supplementary materials, some heuristic elements of answer based on information theory are given, but this is a mathematical question deserving further theoretical investigation and likely to be extremely challenging. The goal here, though more modest, is also more urgent at the time: to broaden the wealth of possible designs that can be used and determine the ones that are the most effective among them, in order to provide quickly implementable and efficient strategies.

# 2 Material and Methods

## 2.1 Scope of the study and constraints

The present study focuses solely on the *two-rounds group testing* with the "trivial algorithm" [Kni95], i.e. aliquots are drawn from samples to be tested, then gathered in a specific way (the *design*) into pools which are tested in a first round. An idealized framework is considered, assuming no testing error; a sample is then known to be negative if it is included in any pool testing negative; a sample is known to be positive if it is contained in any pool testing positive all of whose other samples are already known to be negative; all other samples are of

*Univ Paris Est Creteil, Univ Gustave Eiffel, CNRS, LAMA, F-94010 Creteil, France
benoit.kloeckner@u-pec.fr
†Ecole Normale Superieure de Lyon, France
pierrick.tolle@ens-lyon.fr

unknown status after the first round. The second round consists in testing all of those samples whose positivity was left unknown.

E.g. in Dorfman's single-pool designs, $N$ samples are all contained in 1 pool. The first round consists in a single test; if the pool is tested negative, then there is no need for a second round since all samples are known to be negative, but if the pool is tested positive, then all samples are of unknown status and must be tested in the second round. The most interesting designs have several pools overlapping each other, so that samples have aliquots in several pools. This overlaps both increase the efficiency and introduce some redundancy, allowing potential detection of testing errors. Only the former aspect is considered here, while the latter deserves further study.

Several parameters are important in a design:

- the total number $N$ of samples,
- the number $M$ of pools,
- the maximal size $K$ of pools,
- the maximal number $L$ of different pools containing any given sample.

For practical applicability, designs are required to satisfy the constraints

$$K \leq 100 \tag{1}$$
$$N \leq 1000 \tag{2}$$

The first constraint aims at limiting the decrease in sensitivity of RT-PCR due to the dilution of many aliquots in a pool; the value 100 is the highest value found recorded in the literature for which it is known to be possible to detect a single positive aliquots in a pool [MNB+20]. The most efficient designs in the experiments turn out to have much lower $K$, so that using such a high bound actually adds to the strength of the present results: it means these designs are efficient even when including competing designs with high $K$.

The second constraint limits the number of samples to be gathered before creating the pools for the first round. It is easy to produce extremely efficient designs at small positivity rate if one allows much larger values of $N$, but gathering millions of samples in a single testing facility before starting the first round is obviously impractical. Note that the few designs found in the literature for SARS-CoV-2 group testing that would be excluded by constraint (2) are also ruled out by (1).

The selection of designs resulting from these constraints are presented in Supplementary material (Section 6), and the most efficient ones will be described briefly below.

## 2.2 Measure of efficiency

The quantity used to quantify the efficiency of design is the *expected number of test per sample* $\mathrm{ETS}(D, p)$, which depends on the design $D$ and the positivity rate $p \in (0, 1)$. It is assumed that all samples have the same probability $p$ of being positive and are independent of each other.[1] Since the number of pools $M$ is also the number of test performed in the first round, $\mathrm{ETS}(p, D) = (M + \mathbb{E}(U))/N$ where $U$ is the number of unknown samples at the end of the first round, which is random and depends on both $D$ and $p$.

One could want to penalize the tests performed in the second round, since they incur a delay. However, to avoid long delays it is crucial to ensure the flow of tests to be tested does not outruns the available capacity; since that flow is measured by $\mathrm{ETS}(D, p)$, it is kept it as measure of efficiency. To ensure that there are few samples delayed to the second round, designs are further required to be 1-*perfect*, i.e. to be able to detect and pinpoint a single positive sample among the $N$ tested (see Sections 5.3.6 and 6.8 in the Supplementary material), with an exception: Dorfman's single-pool designs are not 1-perfect, but are kept as comparison point.

For each tested design and each value of $p$, 100 000 independent *in silico* experiments have been performed, determining how many tests would be needed for a pseudo-random population of $N$ individuals (see Supplementary material, Section 5.4); the average number of test per sample is used as estimate for $\mathrm{ETS}(p, D)$, with two digits confidence thanks to the large number of repetitions.

All computations where performed and data generated in Sagemath [The20]; source codes are provided and can be run in Python.

## 2.3 Optimality and relevance of designs

For each $p$ in the tested range, first the design $D_0(p)$ with the smallest (estimated) $\mathrm{ETS}(p, D_0)$ has been determined; then all designs performing almost as well, precisely all those with $\mathrm{ETS}(p, D) \leq 1.02\,\mathrm{ETS}(p, D_0(p))$ have been selected. Among those, designs that where outperformed by another selected design having fewer samples where discarded, so as to eliminate unnecessarily complex designs. To each remaining designs are associated two ranges of $p$: the *optimality range*, for which $\mathrm{ETS}(p, D)$ is at most 2% above $\mathrm{ETS}(p, D_0(p))$; and the *relevance range*, on which $\mathrm{ETS}(p, D)$ is at most 10% above $\mathrm{ETS}(p, D_0(p))$. The point of this larger range is that it might be unpractical to fine tune the choice of design as the measured positivity rate changes, e.g. because of uncertainty on $p$, and it could be more convenient to trade some efficiency (here, up to 10%) for stability or simplicity of the design.

# 3 Results

Table 1 gathers the results, listing all designs that are optimal at some value of $p$ in the sense defined above.

[1] Lack of independence can actually improve group testing. [BSS+22]

Among them, as far as we know only *Steiner triple systems* had already been proposed, and even inside this family no systematic comparison had been done to know how to adjust the choice of parameter depending on the positivity rate.

*Duals of complete graphs* turn out to be extremely efficient in two ranges of positivity rate: for $p \geq 7\%$ and for $p \leq .4\%$. It would be expected from [MT11] that for small $p$, the best designs should have samples included in more than 2 pools, but the constraint (2) on $N$ rules out these theoretically efficient designs, see Section 7.2 in Supplementary material.

Observe that if one seeks for simplicity to cover a large range (say $p \leq 14\%$) with few different designs, using the "relevance" ranges only 4 designs are needed, e.g. $K^*_9$, 4-BIBD$^*_{85}$, STS$^*_{73}$ and $K^*_{44}$.

While in the 1.2% – 1.6% range the best designs have a rather large number of pools, if one wants to restrict to designs with $M \leq 93$ to fit in a 96-wells plate with 3 controls, one can still cover all values of $p$ by paying a small efficiency cost, using 4-BIBD$^*_{88}$ and STS$^*_{67}$ instead of 4-BIBD$^*_{97}$, 4-BIBD$^*_{100}$ and 4-BIBD$^*_{109}$.

## 3.1 Description of optimal designs

Let us briefly describe the designs that turned out optimal at some range; more details are given in the Supplementary material.

Dorfman's *single-pool* designs are considered as a standard comparison point. This family is named `single` in the computer files, while its individual members are denoted by SP$_n$ (where the parameter $n$ corresponds here to the size of the pool).

The family `complete_n` of *dual of complete graphs* is a new proposal of this work, and proves both simple and efficient both at medium-high and low positivity rate. The member of parameter $n$ in this family is denoted by $K^*_n$ in accordance with mathematical habits, and can be described as follow: picture $n$ points, and join every pair of points by an edge. Then each edge represents a sample, each points represents a pool, and each sample has aliquots in exactly 2 pools, one for each ends of the edge represented it. Figure 1 shows $K^*_4$.

The family `steiner3_n` is a selection of particular *dual of Steiner Triple System*, denoted by STS$^*_n$ where $n$ is subject to constraints; in these design, each sample has aliquots in 3 pools.

The family `BIBD41_v` is a selection of particular *dual of* $(v, 4, 1)$*-Balanced Incomplete Block Designs*, denoted by 4-BIBD$^*_v$, where each sample has aliquots in 4 pools.

Last, there is one member of the family `spc_nm` of *square products of duals of complete graphs*, obtained by an algebraic operation combining two designs $K^*_n$ and $K^*_m$, where each sample has again aliquots in 4 pools.

## 3.2 Comparisons with the literature

### 3.2.1 Single-pool design

Figure 2 compares for each $p$ the performance of the best of Dorfman's single-pool design with the best design selected in Table 1. There are two ranges where SP$_n$ is optimal: for $p \gtrsim 12\%$, with $n = 3$, and for very small $p$ ($< .1\%$) with large $n$: the simplicity of this design is efficient in high $p$, and also enables one to satisfy constraint (2) even for high values of the parameter $n$ without even approaching the limit imposed on $N$. However, it has the drawback to incur a high number of second-round tests, a consequence of the lack of 1-perfectness (Supplementary material, Sections 6.8 and 5.3.6).

### 3.2.2 matrix designs

Figure 3 compares the performance of *matrix designs* with the designs selected in Table 1. Matrix designs are obtained by representing samples by the cells of a $n \times n$ matrix and defining a pool for each row and a pool for each column of that matrix, and are among the most classical designs.

Matrix designs' ETS are within a factor 1.02 of the lowest one in the 7%–14% range (and within a 1.1 factor around $p = .5\%$), but are not optimal because they are slightly out-performed by simpler designs, notably duals of complete graphs.

### 3.2.3 Hypercubes

Comparison between hypercubes (introduced in [MNB+20]) and the most efficient designs is given in Figure 4. Hypercubes are quite far from optimality in the whole range of $p$, doing best at $p = .4\%$, where they are about 17% less effective than the best achieving designs. Note that the parameter $d = 6$ is excluded: it would be theoretically quite effective at small $p$, but does not satisfy constraint (1) since each pool has $K = 281$ samples.

This lack of efficency of hypercubes is likely to follow from the large intersections between pools; if as little as two samples are positive, there can be up to $2^d$ samples of unknown status after the first round. It could be that a different strategy with more rounds would make them more efficient than other designs, but they should be compared to the designs of Table 1 in such a setting.

### 3.2.4 P-Best and Tapestry

Let us now address two works for which similar comments can be made, [GRK+20] and [SLW+20] introducing respectively the *Tapestry* and *P-BEST* methods. Both are non-adaptative methods, hence are not directly comparable to two-round pooled testing; they necessarily incur the risk of error in diagnosis, and their point is to try and minimize that risk. However both are based on specific designs, which are included to observe how efficient they are for two-round pooled testing.

| Identification | | | Size | | | Range of $p$ for | | Sample performance | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Design | Family | Key | $N$ | $M$ | $K$ | optimality | relevance | $p$ | ETS | Bound |
| $SP_3$ | `single` | `3` | 3 | 1 | 3 | 12% – 30% | 9% – 30% | 20%<br>14% | .82<br>.70 | .72<br>.58 |
| $K_6^*$ | `complete_n` | `6` | 15 | 6 | 5 | 10% – 15% | 7% – 26% | 13% | .67 | .56 |
| $K_7^*$ | `complete_n` | `7` | 21 | 7 | 6 | 8% – 14% | 6% – 21% | 10% | .57 | .48 |
| $K_8^*$ | `complete_n` | `8` | 28 | 8 | 7 | 7% – 12% | 5% – 17% | 8% | .50 | .43 |
| $K_9^*$ | `complete_n` | `9` | 36 | 9 | 8 | 6% – 9% | 4% – 14% | 6% | .42 | .36 |
| $STS_{25}^*$ | `steiner3_n` | `25` | 100 | 25 | 12 | 5% – 6% | 4% – 9% | 5% | .36 | .32 |
| $STS_{27}^*$ | `steiner3_n` | `27` | 117 | 27 | 13 | 4% – 6% | 2.9% – 8% | 4% | .31 | .28 |
| $STS_{33}^*$ | `steiner3_n` | `33` | 176 | 33 | 16 | 2.9% – 4% | 2.3% – 6% | 3% | .26 | .23 |
| $STS_{37}^*$ | `steiner3_n` | `37` | 222 | 37 | 18 | 2.8% – 3% | 2.1% – 4% | 2.8% | .24 | .22 |
| $STS_{39}^*$ | `steiner3_n` | `39` | 247 | 39 | 19 | 2.6% – 3% | 1.8% – 4% | 2.6% | .23 | .21 |
| 4-$BIBD_{73}^*$ | `BIBD41_v` | `73` | 438 | 73 | 24 | 2.2% – 3% | 1.8% – 4% | 2.4% | .21 | .19 |
| 4-$BIBD_{76}^*$ | `BIBD41_v` | `76` | 475 | 76 | 25 | 2.1% – 3% | 1.7% – 3% | 2.2% | .20 | .18 |
| 4-$BIBD_{85}^*$ | `BIBD41_v` | `85` | 595 | 85 | 28 | 1.8% –2.5% | 1.4% – 3% | 2% | .19 | .17 |
| 4-$BIBD_{88}^*$ | `BIBD41_v` | `88` | 638 | 88 | 29 | 1.7% – 2.4% | 1.4% – 3% | 1.8% | .17 | .16 |
| 4-$BIBD_{97}^*$ | `BIBD41_v` | `97` | 776 | 97 | 32 | 1.5% – 2.1% | 1.1% – 2.7% | 1.6% | .16 | .14 |
| 4-$BIBD_{100}^*$ | `BIBD41_v` | `100` | 825 | 100 | 33 | 1.4% – 2% | 1.1% – 2.6% | 1.4% | .15 | .13 |
| 4-$BIBD_{109}^*$ | `BIBD41_v` | `109` | 981 | 109 | 36 | 1.2% – 1.8% | .9% – 2.3% | 1.2% | .13 | .12 |
| $STS_{67}^*$ | `steiner3_n` | `67` | 737 | 67 | 33 | 1% – 1.1% | .6% – 1.6% | 1% | .12 | .10 |
| $STS_{73}^*$ | `steiner3_n` | `73` | 876 | 73 | 36 | .7% – 1.1% | .5% – 1.4% | .7% | .098 | .076 |
| $K_7^* \boxtimes K_{10}^*$ | `spc_nm` | `(7, 10)` | 945 | 70 | 54 | .5% – .7% | .4% – 1% | .5% | .084 | .058 |
| $K_{38}^*$ | `complete_n` | `38` | 703 | 38 | 37 | .4% | .3% – .6% | .4% | .076 | .048 |
| $K_{43}^*$ | `complete_n` | `43` | 903 | 43 | 42 | .3% – .4% | .1% – .5% | .3% | .064 | .038 |
| $K_{44}^*$ | `complete_n` | `44` | 946 | 44 | 43 | .1% – .4% | .1% – .5% | .1% | .049 | .015 |
| $K_{45}^*$ | `complete_n` | `45` | 990 | 45 | 44 | 0 – .4% | 0% – .5% | $\to 0$ | $\to .045$ | $\to 0$ |
| $SP_{100}$ | `single` | `100` | 100 | 1 | 100 | $\ll .1\%$ | $\ll .1\%$ | $\to 0$ | $\to .01$ | $\to 0$ |

Table 1: *List of optimal designs.*
Identification include the name used in the present paper to denote the design, as used in Section 6, the name of the family as per the source code used and provided in Supplementary material, the key specifying the design inside its family (to be entered as a string). Three of the parameters measuring the complexity of the design are given: the number of samples $N$, the number of pools $M$ and the size of pools $K$. The optimality range is the set of $p$ at which the design is at worst 1.02 times more costly than the absolute best of the experiments, and the relevance range is the set of $p$ at which the design is at worst 1.1 times more costly than the best. Last, performance at one or two value of $p$ is provided alongside the best known theoretical lower bound for two-stage pool testing (this bound does not include the constraints (1) and (2)).

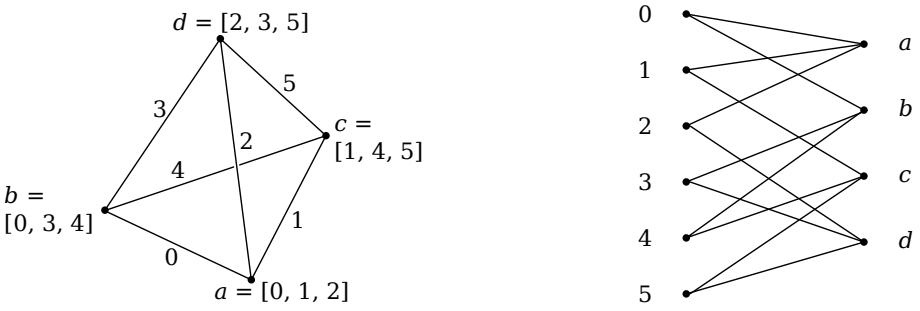


Figure 1: graphical descriptions of $K_4^*$. Left: the complete graph $K_4$, its dual has samples $0, 1, 2, 3, 4, 5$ and pools $a, b, c, d$; right: bipartite graph representation, where each edge means that the sample at its left end has an aliquot in the pool at its right end.

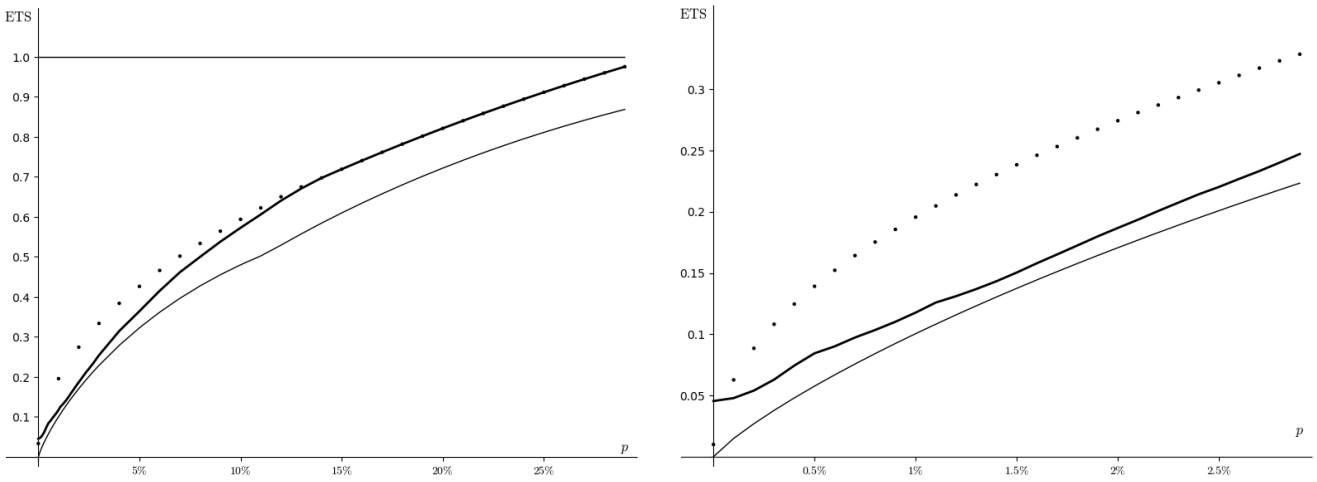


Figure 2: Comparison between the best **single-pool** designs (dots) and the best identified design (thick curve), for each value of the positivity rate $p$ ; the theoretical lower bound is represented by the solid thin curve. Left: $p \leq 30\%$, right: $p \leq 3\%$

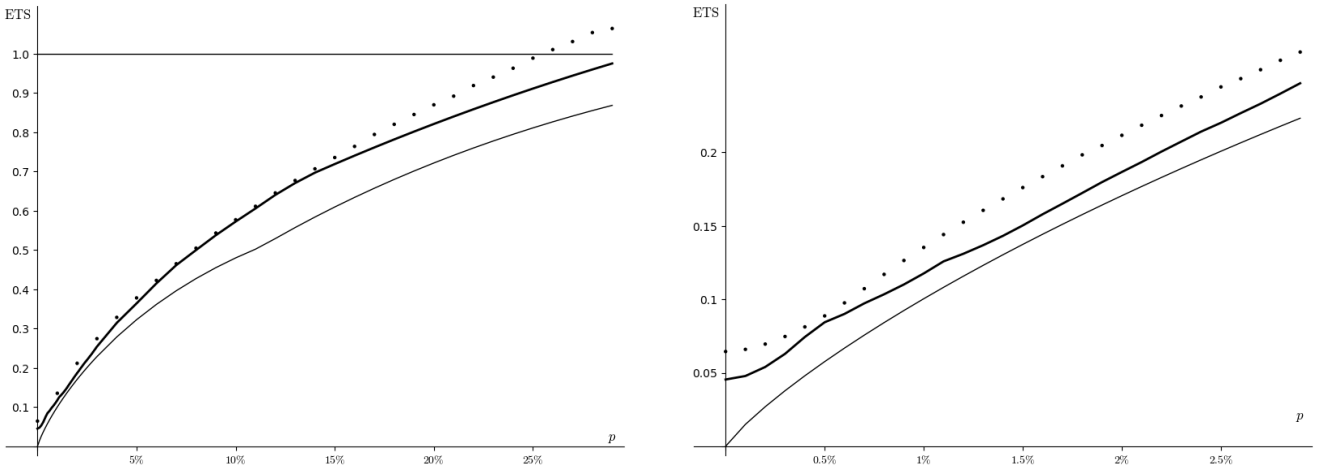


Figure 3: Comparison between the best **matrix** designs (dots) and the best identified designs (thick curve).

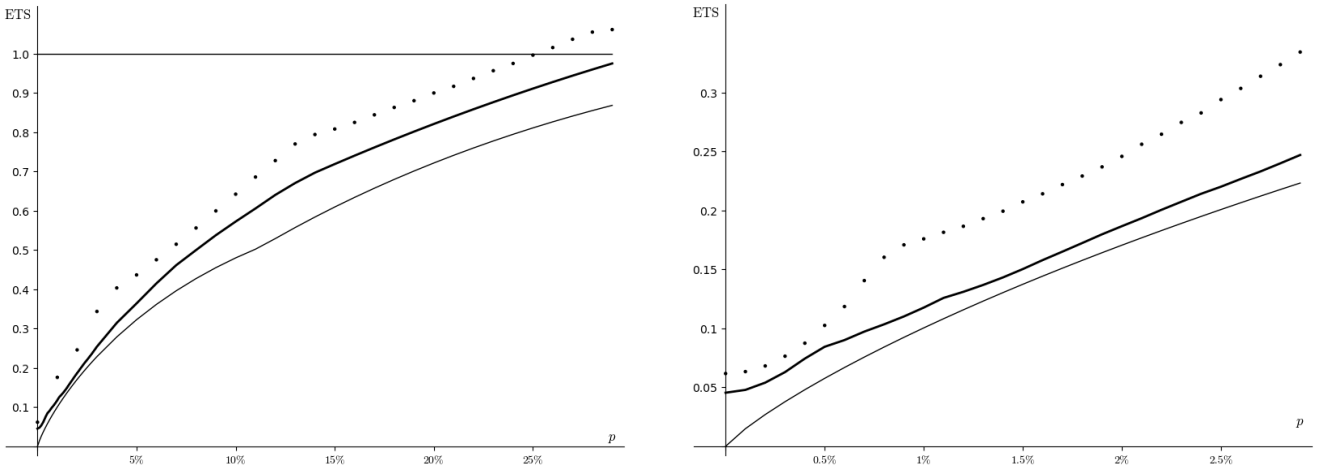


Figure 4: Comparison between the best **hypercube** designs (dots) and the best identified designs selected (thick curve). The thin curve gives the best known lower bound.

Conversely, other designs that prove more efficient than theirs in two-round pooled testing might also be more effective in their respective methods.

While Tapestry's Steiner-based design is not optimal, being outperformed by simpler designs, it is within a factor 1.05 of the best designs in the range 1% – 1.4%, see Figure 5. The performances given in Table 1 of [GRK+20] are even better than allowed by the information bound for the given number of tests, if those tests where binary. The point is that by taking into account the quantitative result of each RT-PCR, they manage to extract several bits of information for each pool. It would thus be interesting to see how far the Tapestry method could go with e.g. dual of complete graphs as designs.

The design used in P-Best is based on an error correcting code and performs relatively well in the range 1% – 1.5%, but not as close to optimality as Tapestry's, see Figure 6. Again, it would be interesting to see whether the P-Best method can gain from being used with the best-performing designs found in the present article.

## 4 Discussion

An extensive comparison of theoretical performances of two-round group testing was performed for a broad variety of designs, enabling the identification of the most efficient design at each positivity rate from .1% to 30%. The new family of dual of complete graphs has been introduced, outperforming all previously proposed designs in the ranges $p \simeq 7\%$–$13\%$ and $p < .4\%$.

Comparing for a given $p$ the performance of the best tested design to the best design among those previously described (single-pool, matrices, hypercube, P-Best, Tapestry), the gain in performance is often quite small (e.g. of the order of 1% at $p = 6\%$). However, this study still gives important information to optimize efficiency of group testing.

First, in the regime of small $p$, better performance enhancement is achieved, e.g. for $p = .5\%$ the design $K_7^* \boxtimes K_{10}^*$ achieves an ETS of .0844, while the best of previous design is the $28 \times 28$ matrix, with an ETS of .0888, a relative improvement of almost 5%.

Second, in many cases the slightly more efficient designs selected as optimal are quite simpler than the previous best-performing ones. For example, at $p = 6\%$, while $K_9^*$ performs only modestly better than the previous best-performing design (the $8 \times 8$ matrix design), it has only 36 samples and 9 pools of 8 while the matrix design has 64 samples and 16 pools of 8. This makes the former more flexible, as one has less samples to gather before starting pooling them. The number of pools is especially kept low by dual of complete graphs for very small $p$, leaving the possibility of testing up to two first rounds simultaneously on a typical 96-wells plate, while still letting room for controls.

Last and most importantly, by comparing a large variety of designs the present experiments provide a precise information of which known design is best performing at each positivity rate. Previously only quite rough or partial information was available (e.g. [MNB+20] gave such information, but only inside the hypercube family): what was mentioned as "the previous best-performing design" was not known to be so. The information provided can thus be used to tune two-round group-testing with the observed positivity rate, and the optimal designs identified here can be tested in other methods such as P-Best, Tapestry, etc.

Many interesting questions are left open:

- find designs approaching the theoretical bound for $p \gtrsim 5\%$,
- improve the known lower bound for two-round testing under constraints (1) and (2) for small $p$,
- study the effect of testing error for the designs pointed out as optimal.

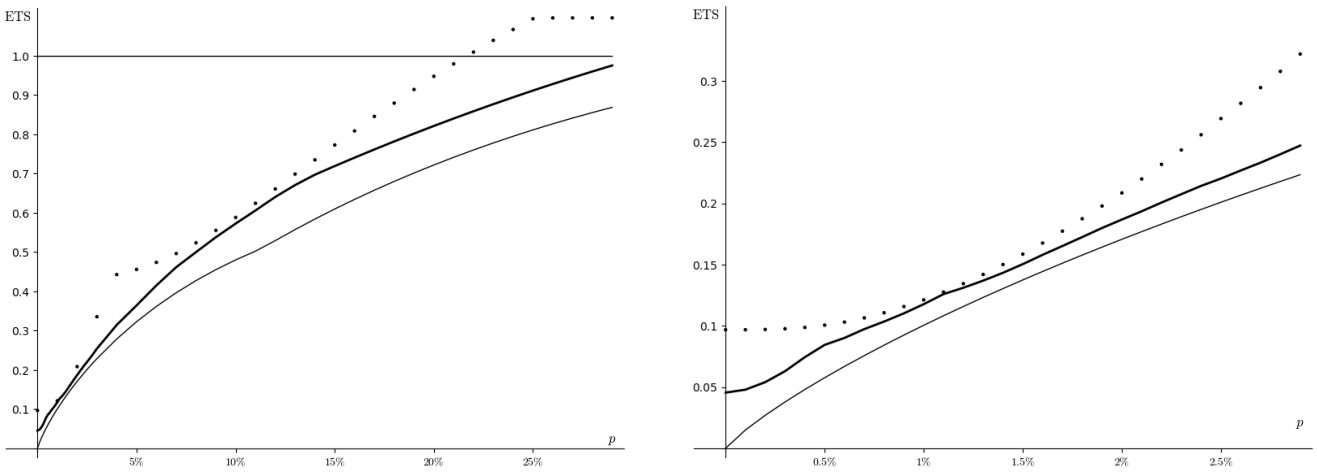


Figure 5: Comparison between the best **Tapestry** designs (dots) and the best identified designs (thick curve).

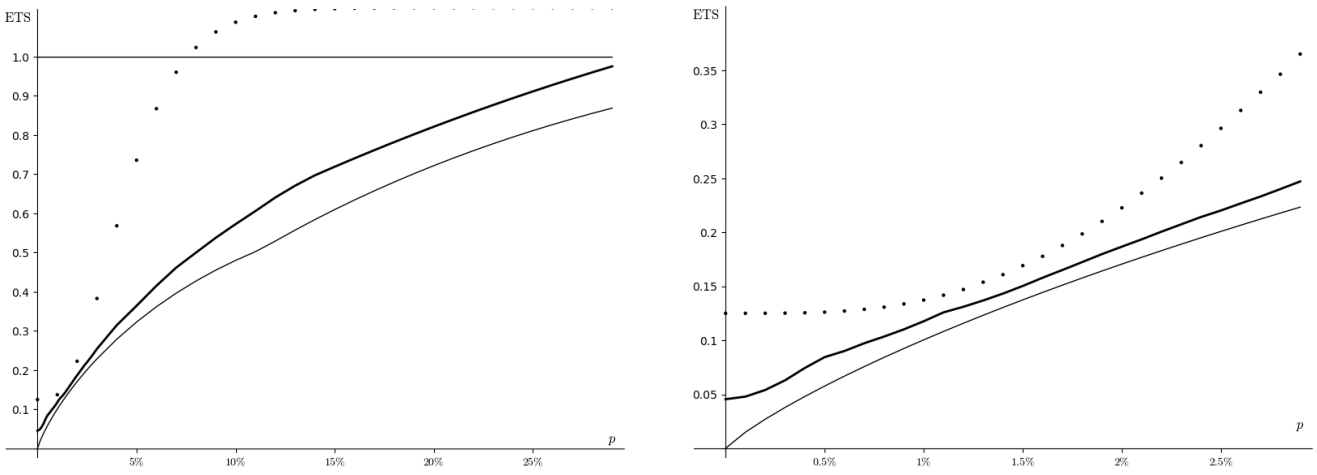


Figure 6: Comparison between the **P-Best** design (dots) and the best identified designs (thick curve).

# References


[ACFSL20] Diego Aragón-Caqueo, Javier Fernández-Salinas, and David Laroze. Optimization of group size in pool testing strategy for sars-cov-2: A simple mathematical model. *Journal of Medical Virology*, 92(10):1988–1994, 2020. 1

[AH97] Ian Anderson and Iiro Honkala. A short course in combinatorial designs. Internet Edition, Spring, 1997. 6.3

[BMR21] Vincent Brault, Bastien Mallein, and Jean-François Rupprecht. Group testing as a strategy for covid-19 epidemiological monitoring and community surveillance. *PLoS Comput Biol*, 17(3):e1008726, 2021. 1

[BSS+22] Leonardo J Basso, Vicente Salinas, Denis Sauré, Charles Thraves, and Natalia Yankovic. The effect of correlation and false negatives in pool testing strategies for COVID-19. *Health Care Management Science*, 25(1):146–165, 2022. 1

[DHH00] Dingzhu Du, Frank K Hwang, and Frank Hwang. *Combinatorial group testing and its applications*, volume 12 of *Series on Applied Mathematics*. World Scientific, 2000. 5.3.6

[Dor43] Robert Dorfman. The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4):436–440, 1943. 1

[GAR+21] Sabyasachi Ghosh, Rishi Agarwal, Mohammad Ali Rehan, Shreya Pathak, Pratyush Agarwal, Yash Gupta, Sarthak Consul, Nimay Gupta, Ritesh Goenka, Ajit Rajwade, et al. A compressed sensing approach to pooled rt-pcr testing for covid-19 detection. *IEEE Open Journal of Signal Processing*, 2:248–264, 2021. 5.3.5, *iv*

[GRK+20] Sabyasachi Ghosh, Ajit Rajwade, Srikar Krishna, Nikhil Gopalkrishnan, Thomas E Schaus, Anirudh Chakravarthy, Sriram Varahan, Vidhya Appu, Raunak Ramakrishnan, Shashank Ch, et al. Tapestry: A single-round smart pooling technique for covid-19 testing. MedRxiv, 2020. 1, 3.2.4, *iv*

[HDFA+21] Victoria Haldane, Chuan De Foo, Salma M Abdalla, et al. Health systems resilience in managing the covid-19 pandemic: lessons from 28 countries. *Nature Medicine*, 27(6):964–980, 2021. 1

[Kni95] Emanuel Knill. Lower bounds for identifying subset members with subset queries. In *SODA*, pages 369–377, 1995. 2.1, 5.1, *iii*

[LPBG+20] Stefan Lohse, Thorsten Pfuhl, Barbara Berkó-Göttel, Jürgen Rissland, Tobias Geißler, Barbara Gärtner, Sören L Becker, Sophie Schneitler, and Sigrun Smola. Pooling of samples for testing for sars-cov-2 in asymptomatic people. *Lancet Infect Dis.*, 20(11):1231–1232, 2020. 5.3.1

[LTM+21] Nefeli Lagopati, Panagiota Tsioli, Ioanna Mourkioti, Aikaterini Polyzou, Angelos Papaspyropoulos, Alexandros Zafiropoulos, Konstantinos Evangelou, George Sourvinos, and Vassilis G Gorgoulis. Sample pooling strategies for sars-cov-2 detection. *Journal of virological methods*, 289:114044, 2021. 1, *ii*

[MNB+20] Leon Mutesa, Pacifique Ndishimye, Yvan Butera, Jacob Souopgui, Annette Uwineza, Robert Rutayisire, Ella Larissa Ndoricimpaye, Emile Musoni, Nadine Rujeni, Thierry Nyatanyi, et al. A pooled testing strategy for identifying SARS-CoV-2 at low prevalence. *Nature*, pages 1–8, 2020. link. 1, 2.1, 3.2.3, 4, 5.3.1, 5.3.4, *iii*

[MSLH+21] Alexandra Martin, Alexandre Storto, Quentin Le Hingrat, Gilles Collin, Barbara André, Allison Mallory, Rémi Dangla, Diane Descamps, Benoit Visseaux, and Olivier Gossner. High-sensitivity sars-cov-2 group testing by digital pcr among symptomatic patients in hospital settings. *J Clin Virol.*, 141:104895, 2021. 1

[MT11] Marc Mézard and Cristina Toninelli. Group testing with random pools: Optimal two-stage algorithms. *IEEE Transactions on Information Theory*, 57(3):1736–1745, 2011. 3, 5.3.2, 5.3.6, *iii*, *vi*, *vi*, 7.1, 7.1

[SAKH20] Nasa Sinnott-Armstrong, Daniel L Klein, and Brendan Hickey. Evaluation of group testing for sars-cov-2 rna. MedRxiv, 2020. 1, *ii*

[SLW+20] Noam Shental, Shlomia Levy, Vered Wuvshet, Shosh Skorniakov, Bar Shalem, Aner Ottolenghi, Yariv Greenshpan, Rachel Steinberg, Avishay Edri, Roni Gillis, et al. Efficient high-throughput SARS-CoV-2 testing to detect asymptomatic carriers. *Science advances*, 6(37), 2020. 1, 3.2.4, *iv*

[Smi04] Clayton Smith. On the existence of $(v, 5, 1)$-bibds. `https://argilo.net/files/bibd.pdf`, 2004. 6.4

[Sti07] Douglas Stinson. *Combinatorial designs: constructions and analysis*. Springer, 2007. 6.4

[The20] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 9.0)*, 2020. `https://www.sagemath.org`. 2.2, 6.3

[VP20] Damir Vukičević and Ozren Polašek. Optimizing the diagnostic capacity for covid-19 pcr testing for low resource and high demand settings: The development of information-dependent pooling protocol. *Journal of global health*, 10(2), 2020. 1

# Supplementary material

## 5 Justification of choices and constraints

Let us justify in more detail the restriction to two-round group testing, the choice of ETS as measure efficiency, and the restriction on considered designs.

### 5.1 First restriction: two-rounds group testing

Group testing is usually divided into *adaptative* and *non-adaptative* procedures. Non-adaptative group testing proceeds in a single rounds, all pools being tested in parallel. It is not possible to obtain in this way a complete information on the positivity of each sample unless we use as many tests as the number of samples, which defeats the purpose of group testing. One thus has to accept a possibility of error (on top of all other sources of error due to the fallibility of the tests themselves) and will try to minimize that risk. Adaptive group testing proceeds in several consecutive rounds, the pools of a round being decided depending on the results of the previous rounds. Two issues appear: first the logistic is complicated as one needs to take subsequent aliquots in certain samples to build pools at each round; second the results can be delayed a lot if many consecutive rounds are needed. Two-round group testing as defined in Section 2 has been identified as a balanced middle ground [Kni95], and this why we focus on this method. It is likely that some of our results could be expanded to other variants of group testing: choices of pools that are efficient for two-rounds group testing may also be efficient for other procedures.

### 5.2 Designs for two-rounds group testing

The restriction to two-rounds group testing still leaves many ways to proceed: one has to choose how to organize the pools. This corresponds to a combinatorial structure know in mathematics as a *hypergraph* or *incidence structure*, and sometimes presented as an "algorithm" for group testing. Formally, a hypergraph can be presented in many ways. One of the most common is to give a set of *vertices* $V$ and a set $E$ of subsets of $V$ called *edges*. Formally, what we call a *design* is simply a hypergraph for which $|E| < |V|$ (we use $|\cdot|$ to denote number of elements); if $D = (V, E)$ is a design, elements of $V$ thus represent the samples to be tested and elements of $E$ represent the pools (also called groups) that will be used in the first round, so that the above condition means that there shall be less pools than samples. One reason to use a specific name for designs is that many hypergraphs that have been constructed in the mathematical field of combinatorics have more edges than vertices, but still yield interesting designs through *duality*, an operation that we will explain later on.

### 5.3 Design parameters and constraints

Several parameters of a design $D = (V, E)$ have to be constrained to allow the two-rounds group testing to be realized in practice. These constraints may vary depending on what is being tested, and we focus here on SARS-CoV-2 detection through RT-PCR tests.

#### 5.3.1 Size of pools

First, large pools induce a reduction in RT-PCR sensitivity: a single positive sample has to be detected despite its dilution even when all other samples tested in the pool are free of SARS-CoV-2. The first parameter of interest is thus the maximal size of pools

$$K := \max\{|g| : g \in E\}.$$

This parameter influences the risk of false negative in a gradual way: the lesser $K$, the lesser the loss in sensitivity. We draw the line at the highest value we could ensure has been checked to still be viable in practice, namely

$$K \leq 100 \tag{1}$$

where the value 100 is reported in [MNB+20]. Lesser values where validated in other works, e.g. 30 in [LPBG+20]. While we ruled out designs with higher values of $K$, our raw data can easily be used to determine good designs with lower values of $K$, e.g. for situations where positive samples are expected to have low viral load.

#### 5.3.2 Regularity

While this is not a strict constraint, most of the designs we consider and all of the optimal ones are *regular*, by which we mean that all pools have the same size and all samples appear in the same number of pools. Many mathematical constructions of hypergraphs are regular. Moreover the size of pools is known to be an important parameter to tune depending on $p$, see e.g. [MT11], and if some pools are significantly larger than others, they cannot be both of optimal size.

As counter examples, some matrix designs with a different number of rows and columns (thus not regular) have been sometimes used in the literature, but we restricted to regular matrix designs.

#### 5.3.3 Overall size

All tests of the first round are to be run in parallel to avoid unnecessary delays, and to build the pools all samples have to be available in the preparation phase of the first round. For small $p$, we could easily find designs

that are close to the theoretical limit in efficiency but need to manage millions of samples together, i.e.

$$N := |V| > 10^6$$

This is obviously impractical, and we choose the following constraint:

$$N \leq 1000 \tag{2}$$

Let us add an argument to motivate constraint (2) on top of the one given in Section 2: considering that tests are performed locally to avoid difficult logistic issues, a medium city of $\sim 100,000$ inhabitants where each day 2% of the population is tested would gather enough samples to perform a two-rounds group testing procedure with $N = 1000$ twice each day. Any larger $N$ would imply gathering delays over half a day in such a situation.

#### 5.3.4 Number of aliquots per sample

Another parameter that should in principle be controlled is the maximal number $L$ of pools a sample is tested in, i.e. the maximal number of aliquots from any sample needed in the first round (an additional aliquot being posibly needed in the second round). Setting $E_s = \{g \in E \mid s \in g\}$ the set of pools containing (an aliquot of) the sample $s$, we thus consider

$$L := \max\{|E_s| \colon s \in V\}$$

However, several dozen of aliquots are easily drawn from a typical sample issued from a nasopharyngeal swab [MNB+20]. We know of no potentially effective design that satisfies constraints (1) and (2) and comes even close to that number (we will see that the optimal designs have $L \leq 4$), making it pointless to constraint $L$.

#### 5.3.5 Number of pools

A last parameter of interest is the number of pools

$$M := |E|.$$

It is proposed in [GAR+21] to add the constraint $M \leq 93$, since it is a standard number of tests that can be run in parallel on a RT-PCR machine (using a 96-wells plate with three wells used as controls). We choose not to include this constraint since bypassing it in practice is a matter of industrial design choice, it does not need new knowledge or technology. However most of the efficient designs turn out to satisfy it, and we gave alternative propositions for the range where they do not.

Even without constraining it, $M$ is important as it influences the efficiency of the design. The first round indeed consist in $M$ tests, so that lesser $M$ is better for very small $p$; and if $M$ is too close to $N$ one cannot expect much savings.

#### 5.3.6 Perfect identification of single positives

We also mostly restricted to designs that are able to detect a single positive sample without a second round, which we call 1-*perfect* designs (contrary to what is sometimes assumed in combinatorial group testing [DHH00], we mean that this detection should be possible without the knowledge that there is exactly one positive sample). This property is shared by all designs found in the literature except Dorfman's single-pool design (which we keep as the only exception to this restriction, as a comparison point and because in some regime it seems difficult to improve upon it). We will see that it also ensures that in the range where a design is most efficient, very few samples need to be tested in the second round.

It is easy to see that a design is 1-perfect if and only if both following conditions are fulfilled:

*i.* every sample is in some pool, and

*ii.* for every two samples $s, t \in V$, there exist a pool $g \in E$ that contains $s$ but not $t$.

*Proof.* Indeed, assume these conditions are satisfied and $s_0$ is the unique positive sample. Then every $s \neq s_0$ will be detected as negative since it belongs to a pool not containing $s_0$, hence not containing any positive sample. Meanwhile, $s_0$ belongs to at least one pool $g$, which will appear positive. But all other samples being known negative, at the end of the first round $s_0$ will be known to be positive.

Conversely, if some sample is in no pool then obviously its status will be unknown at the end of first round; and if there are two samples $s, t$ such that every pool containing $s$ also contains $t$, then the first round cannot distinguish the situation where they are both positive from the situation where $s$ is positive and $t$ negative. □

Let us make an observation that will be used later on. Consider a 1-perfect design where some sample $s$ belongs to only one pool. If this pool contained any other sample $t$, the design would not satisfy *ii*. Therefore this pool contains only $s$, but then removing $s$ and its pool can only improve the design. We can thus restrict 1-perfect designs to the case when *every* sample belongs to at least 2 pools.

One situation ensuring *ii* is when every sample belongs to at least 2 pools *and* every two distinct pools have at most one sample in common (this is equivalent to the condition "having girth at least 6" in [MT11]). This will always be the case in the most effective designs, a phenomenon explained in [MT11] and that should not come as a surprise if we take a information-theoretic approach: while some intersection between pools is necessary to identify some positive samples in the first round, large intersections would make first-round tests all the more redundant. Conversely, smaller intersections ensure more information is gathered in the first round.

### 5.4 Evaluation procedure for $\mathrm{ETS}(p, D)$

To sum-up, we are concerned with finding for each $p$ designs $D = (V, E)$ that are 1-perfect, that satisfy the

constraints (1) and (2), i.e. with no pool grouping more than 100 samples and with no more than 1000 samples overall, and which make $\mathrm{ETS}(D,p)$ as small as we can.

While for simple $D$ we could compute explicit expressions for the function $p \mapsto \mathrm{ETS}(D,p)$, even mildly complex designs make this task daunting, hence the use of *in silico* experiments, which we now describe in detail. Given a number $p$ and a design $D$, an experiment consists in the following steps:

*i.* draw a pseudo-random vectors with $N$ entries in $\{0,1\}$ (0 for samples not carrying SARS-CoV-2, 1 for carriers) that are independent and have probability $p$ of taking the value 1, representing the population of samples to be tested,

*ii.* assign to each pool a value 1 or 0 depending whether it contains a positive sample or not,

*iii.* construct a new $N$-vector representing the inference on the population that can be drawn from the pool values: a sample is given the value 0 (sure negative) if it belongs to at least one negative pool, the value 1 (sure positive) if it belongs to at least one positive pool where all other samples are tagged as sure negatives, and the value '?' (unknown) in all other cases, as in [Kni95, MT11],

*iv.* count the number $U$ of '?' values in that vector,

*v.* return the number $M+U$ of tests needed to determine the status of all samples over the two rounds.

For each $D$ and each value of $p$ we repeat this experiment $100,000$ times and use the value $\frac{M+\bar{U}}{N}$ as estimator for $\mathrm{ETS}(D,p)$, with $\bar{U}$ the average of $U$ over all repetitions.

We consider the following values of $p$: first a range of rather large values $\{1\%, 2\%, \dots, 29\%\}$, then a finer range $\{.1\%, .2\%, \dots, 2.9\%\}$ for small values. For even smaller value, one will be mostly reduced to compare Dorfman's single pool method to the best designs for $p = .1\%$: it seems indeed difficult, if not outright impossible to improve on them while satisfying constraint (2), see Section 7.2.

Let us evaluate the precision of our estimation. The estimator for $\mathrm{ETS}(D,p)$ contains a constant term $M/N$ and a variable term $\bar{U}/N$; given the number of independent repetitions, we can assume $\bar{U}$ has a Gaussian distribution. Since $U$ compounds for all samples $s \in V$ of the design the probability that $s$ is tested in the second round, $\bar{U}$ is the sum of $N$ binomial, almost Gaussian random variables, non independent but identically distributed since our designs are symmetric. The variance of $\bar{U}/N$ is not greater than the variance of each of these binomials. Each $s$ provides at each repetition a Bernoulli random variable, with probability of being positive close to $\bar{U}/N$ and thus variance close to $\bar{U}/N(1-\bar{U}/N)$. We can thus estimate the 95% confidence interval for $\mathrm{ETS}(D,p)$ as

$$\frac{M+\bar{U}}{N} \pm 1.96\sqrt{\frac{1}{100,000}\frac{\bar{U}}{N}\Big(1-\frac{\bar{U}}{N}\Big)}.$$

We obtain a typical error never larger than .31 percentage points, about .19 percentage points when $\bar{U}/N = 10\%$, and as small as .06 percentage points when $\bar{U}/N = 1\%$. In all cases, we can confidently keep 2 significant digits.

As an example, consider the case of $p = 8\%$ for $D = K_8^*$. Our estimation for the expected number of test per sample is $\mathrm{ETS}(.08, K_8^*) \simeq .5001$, while $M/N = 8/28 \simeq .2857$, so that $\bar{U}/N \simeq .2144$ and the size of either half of the confidence interval is $\simeq .0025$. We can thus assume with 95% confidence that $.4976 \leq \mathrm{ETS}(.08, K_8^*) \leq .5026$, justifying our reported value of .50. Moreover the designs that have been considered not optimal at that value have $\mathrm{ETS}(.08, D) \gtrsim 1.02 \times .50 = .51$, almost 8 standard deviations above the value obtained for $K_8^*$.

# 6 Representation and building of designs

Before presenting in more details the various families of designs we run our experiments on, let us introduce an important tool to generate such families.

## 6.1 Hypergraphs and duality

A hypergraph $H$ can be specified by giving the set $V$ of vertices and the set $E$ of edges (each a subset of $V$), or in several alternate ways. One is its *incidence matrix* $M = (m_{ij})$ where the column index $j$ runs over $V$, the row index $i$ runs over $E$, and $m_{ij}$ is 1 when the edge $i$ contains the vertex $i$, 0 otherwise. Given $(V,E)$ the incidence matrix can easily be recovered, and given an incidence matrix one can easily rebuild $H$ (up to the naming the vertices, i.e. in mathematical term up to isomorphism). Another way to present the same information is to draw a *bipartite graph*, i.e. a triple $(A,B,I)$ where $A, B$ are sets and $I$ is a subset of $A \times B$, i.e. a set of pairs $(a,b)$ where $a \in A$ and $b \in B$. Given $H = (V,E)$, the corresponding bipartite graph is given by $A = V$, $B = E$ and $I = \{(v,e) \in V \times E \mid v \in e\}$. Conversely, given a bipartite graph $(A,B,I)$ one obtains a hypergraph by defining $V = A$; for each $b \in B$, $e_b = \{a \in A \mid (a,b) \in I\}$; and $E = \{e_b \colon b \in B\}$. In other words, instead of having edges given as sets of vertices, we define them as labels and then use $I$ to describe the incidence relation between vertices and edges (see Figure 1, right).

One interest of these constructions is that both make an important operation, *duality*, easy to picture. The *dual* of a hypergraph $H = (V,E)$ with incidence matrix $M$ and bipartite graph $(A,B,I)$ is the hypergraph $H^*$ whose incidence matrix is $M^T$, the transpose of $M$. Equivalently, $H^*$ is the hypergraph whose bipartite graph is $(B,A,I^*)$ where $I^* = \{(b,a) \mid (a,b) \in I\}$. In other words, $H^*$ exchanges the roles of vertices and edges, while the incidence relation only swaps directions. The dual $H^*$ is of course unique only up to isomorphism.

In the mathematical literature, one can find many hypergraphs with few vertices and many edges; duality enables us to build designs out of them, e.g. duals of complete graphs, duals of Steiner triple systems, etc.

## 6.2 Duals of complete graphs

Let $n \in \mathbb{N}$; the *complete graph* $K_n$ is the hypergraph with vertex set $V = \{1, \dots, n\}$ and edge set $E$ given by all subsets of $V$ having exactly 2 elements. We thus have $|E| = \frac{n(n-1)}{2}$, larger than $|V| = n$ as soon as $n \geq 4$. We shall consider the family $(K_n^*)_{4 \leq n \leq 45}$ of duals of complete graphs. The design $K_n^*$ indeed has $N = \frac{n(n-1)}{2}$ samples, smaller than 1000 for $n \leq 45$, and $M = n$ pools. The case $n \leq 3$ is irrelevant as we get at least as many pools as samples.

In $K_n^*$, each sample is included in exactly $L = 2$ pools, while each pool contains $K = n - 1$ samples (in particular the pool size $K \leq 44$ is much lesser than the constraint of 100). Two pools intersect in exactly one sample, ensuring 1-perfectness.

A good way to picture $K_n^*$ is to draw $K_n$ and recall that the segments representing edges are actually samples for $K_n^*$, and vertices of $K_n$ are actually pools for $K_n^*$. Figure 1 shows this for $n = 4$. In our code, by convention $V = \{0, \dots, N-1\}$ and designs are given by a list of list of vertices: for $K_4^*$, we get

```
[[0,1,2], [0,3,4], [1,4,5], [2,3,5]]
```

while its incidence matrix is

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

## 6.3 Duals of Steiner triples system

A hypergraph $H = (V, E)$ is called a *Steiner triple system* (sometimes a *Steiner triple*) of order $n$ whenever it has $n$ vertices, all its edges have exactly 3 elements, and every pair of distinct vertices is contained in exactly 1 edge. As is well-known, these constraints impose that $n$ is of the form $6k + 1$ or $6k + 3$ for some integer $k$, and the number of edges is $\frac{1}{6}n(n-1)$; we restrict to $n \geq 9$ to avoid trivial cases. It turns out that the above constraint is also sufficient to build Steiner triple systems, but note that there can be different Steiner triple systems of the same order. We will use the ones provided in the Sage mathematical software [The20], described in [AH97] page 32. The dual of that Steiner triple system will be denoted by $\mathrm{STS}_n^*$; it has $N = \frac{1}{6}n(n-1)$ samples and $M = n$ pools, all of size $K = \frac{1}{2}(n-1)$, and each sample belongs to $L = 3$ pools. Every two distinct pools intersect in exactly one sample, ensuring $\mathrm{STS}_n^*$ is 1-perfect. We restrict to $9 \leq n \leq 75$, since larger $n$ would entail $N > 1000$. Observe that again, the pools size is at most 37, well below the limit of 100.

## 6.4 Balanced Incomplete Block Designs of higher multiplicity

Steiner triple systems are particular cases of *Balanced Incomplete Block Designs* (BIBD), defined as any hypergraph $H = (V, E)$ such that all edges have the same number $\ell$ of vertices, all vertices are contained in the same number $r$ of edges and all pairs of distinct vertices are contained in the same number $\lambda$ of edges. Non trivial cases have at least as many edges than vertices, and our designs will again be obtained by duality. Every two pools will then have $\lambda$ samples in common, and following the discussion at the end of Section 5.3 we restrict to $\lambda = 1$; and every sample is contained in the same number $L = \ell$ of pools. We actuality met BIBD with $\lambda = 1$ already: complete graphs ($\ell = 2$) and Steiner triple systems ($\ell = 3$). We further tested $\ell = 4$ and $\ell = 5$, and only the former turned out to be optimal in some range of $p$. Since the complexity grows quickly with $\ell$, it seems unnecessary to explore further (see Section 7.2).

More precisely, we considered the following two families. First, $4\text{-BIBD}_n^*$ is the dual of Sage's implemented BIBD with $\ell = 4$ and $\lambda = 1$ (as described in [Sti07]) over $n$ vertices, where $n$ must have the form $12m + 1$ or $12m + 4$ for some integer $m$; we consider $16 \leq n \leq 109$ to avoid triviality and respect the constraints. $4\text{-BIBD}_n^*$ has $N = \frac{1}{12}n(n-1)$ samples, each in $L = 4$ pools. All pools have $K = \frac{1}{3}(n-1)$ samples (again much lesser than 100) and every two distinct pools intersect in exactly one sample, ensuring 1-perfectness.

Second, $5\text{-BIBD}_n^*$ is the dual of Sage's implemented BIBD with $\ell = 5$ and $\lambda = 1$ (as described in [Smi04]) over $n$ vertices, where $n$ must have the form $20m + 1$ or $20m + 5$ for some integer $m$; we consider $25 \leq n \leq 141$. $5\text{-BIBD}_n^*$ has $N = \frac{1}{20}n(n-1)$ samples, each in $L = 5$ pools. All pools have $K = \frac{1}{4}(n-1)$ samples (again much lesser than 100) and every two distinct pools intersect in exactly one sample, ensuring 1-perfectness.

## 6.5 Products of designs

There are several operations that have been devised to produce new hypergraphs out of known ones. Two particularly relevant ones are the *Cartesian product* and the *square product*. Assume $H_1 = (V_1, E_1)$ and $H_2 = (V_2, E_2)$ are hypergraphs. Their Cartesian product $H_1 \square H_2$ and their square product $H_1 \boxtimes H_2$ share the same set of vertices, $V_1 \times V_2$, i.e. a vertex is a pair $(x_1, x_2)$ where $x_1$ is a vertex of $H_1$ and $x_2$ a vertex of $H_2$.

The Cartesian product has an edge for each pair $(e_1, x_2)$ and one for each pair $(x_1, e_2)$ (where $x_i \in V_i$ and $e_i \in E_i$); the edge associated to $(e_1, x_2)$ gathers all vertices $(y_1, x_2)$ where $x_2$ is fixed and $y_1$ runs over $e_1$; the $(x_1, e_2)$ edges are build similarly.

The square product has an edge for each pair of edges $(e_1, e_2)$ (the first of $H_1$, the second of $H_2$), given by the set $e_1 \times e_2$ of all vertices $(y_1, y_2)$ where $y_1$ runs over $e_1$ and $y_2$ runs over $e_2$.

In the case of designs $D_i$ with $N_i$ samples and $M_i$ pools ($i = 1, 2$), the Cartesian product $D_1 \square D_2$ has $N_1 N_2$ samples and $M_1 N_2 + N_1 M_2$ pools, while the square product $D_1 \boxtimes D_2$ has only $M_1 M_2$ pools for the same number of samples. The number $M/N \in (0, 1)$ of first-round-tests per sample is thus summed under Cartesian product and multiplied under square product. If the $D_i$ are regular, with pool size $K_i$ and each sample belonging to $L_i$ pools, then in $D_1 \square D_2$ each sample is in $L_1 + L_2$ pools but this design is not regular when $K_1 \neq K_2$: it has pools of size $K_1$ and pools of size $K_2$. Meanwhile $D_1 \boxtimes D_2$ is regular, with pool size $K_1 K_2$ and each sample in $L_1 L_2$ pools.

Out of the cases we tried, only the square products of duals of complete graphs produced optimal design in some range.

## 6.6 Designs found in the literature on group testing for SARS-CoV-2

Let us now present the designs that we could found being proposed in the previous literature on group testing for SARS-CoV-2.

*i.* Dorfman's single-pool designs are of course the first ones historically. We denote them by $\mathrm{SP}_n$ where $n$ is any integer $\geq 2$; there is $M = 1$ pool of $N = K = n$ samples. Each sample is in $L = 1$ pool, and these designs are not 1-perfect. Rather than experimental data, we used for these designs the exact expression

$$\mathrm{ETS}(\mathrm{SP}_n, p) = 1 + \frac{1}{n} - (1-p)^n.$$

*ii.* The matrix designs are considered in a number of work, e.g. [LTM+21, SAKH20]; the size $n$ matrix design has $N = n^2$ samples corresponding to the cells of a $n \times n$ matrix, each row and each column of the matrix giving an edge. This is also the Cartesian product of two single-pool designs of size $n$; there are $M = 2n$ pools of size $K = n$, and each sample is in $L = 2$ pools. Pools intersect in at most one sample, making the design 1-perfect.

*iii.* Hypercubes $(\mathrm{Hyp}(d))_{d \in \mathbb{N}}$ have been proposed in [MNB+20]; $\mathrm{Hyp}(d)$ has $N = 3^d$ samples arranged as an hypercube $\{0, 1, 2\}^d$, and there are $M = 3d$ pools, one for each "slice" in any of the coordinate: $\{0\} \times \{0, 1, 2\}^{d-1}$, $\{1\} \times \{0, 1, 2\}^{d-1}$, $\{2\} \times \{0, 1, 2\}^{d-1}$, $\{0, 1, 2\} \times \{0\} \times \{0, 1, 2\}^{d-2}$, etc. Pools have size $K = 3^{d-1}$ and each sample is contained in $L = d$ pools. Two pools intersect in either 0 or $3^{d-2}$ samples, and this design can be checked to be 1-perfect.

*iv.* Tapestry's designs [GRK+20] are constructed from particular Steiner triple systems, with some samples left empty (i.e. some columns deleted in the incidence matrix [GAR+21]), while P-Best [SLW+20] designs are based on error-correcting codes. We used the incidence matrices provided by the authors to include them in the comparison, although the methods presented by these articles are different from two-round testing.

## 6.7 Other designs tested

Let us finally quickly present other designs we tested, although they did not turn out to be optimal in any range of the positivity rate.

*i.* The affine designs are mathematical constructs involving both algebra and geometry. They depend on three integer parameters $n, d, q$, where $0 < d < n$ and $q$ is a prime power. We consider the finite field $\mathbb{F}_q$ and build the affine space $\mathbb{F}_q^n$, then consider the incidence structure given by inclusion between points and subspaces of dimension $d$. This can be obtained from Sage's `AffineGeometryDesign` procedure. There are more edges than vertices, so we considered the dual hypergraph as our designs. Note that some turned out to be optimal in some range, but those which did actually coincide with some previous families, e.g. Affine Geometry Designs with $q = 2$ and $d = 1$ are in fact complete graphs.

*ii.* The projective designs are similar, except that we use projective spaces instead of affine spaces. It amounts to only consider subspaces that contain the origin, and to translate all dimensions by one: point are replaced by lines, $d$-subspaces by $(d+1)$-subspaces, and $\mathbb{F}_q^n$ by $\mathbb{F}_q^{n+1}$. Again, we used Sage's already implemented `ProjectiveGeometryDesign`, and again those which where optimal coincide with previous families (Steiner triple systems, when $q = 2$ and $d = 1$).

*iii.* In the same vein but conceptually simpler, one can consider the incidence structure of $k$-subsets against $m$-subsets in a $p$-set (where $1 \leq k < m \leq p/2$). Namely the design has all $m$-subsets as tags for samples and $k$-subset as tags for pools, and each pool tagged $g$ contains precisely the samples tagged $s$ such that $g \subset s$. Pools can have large intersection, and trying to minimize the average intersection and respect the constraints led us to test the cases $m = 2, k = 4, p = 14$ and $m = 1, k = 3, 9 \leq p \leq 20$.

*iv.* For high positivity rate, we tried in several ways to improve on single-pool designs, without success; we tested the family of products $K_{16}^* \square \mathrm{SP}_n$ (where $K_{16}$ is also the Affine Geometric Design with $q = 2$, d= 1 and $n = 4$), then chains where samples are consecutive integers and pools are intervals of given length and given overlap with the preceeding one, e.g. for 7 pools with overlap of size 1

$$[[0, 1, 2], [2, 3, 4], [4, 5, 6], [6, 7, 8], [8, 9, 10], [10, 11, 12], [12, 13, 14]];$$

and last a "compact" design constructed by hand to have $K = 4$ with pools intersecting at most at one sample (we manage to almost achieve $L = 3$ with $N = 13$, $M = 10$ by letting one sample belong to 4 pools, all others to 3).

*v.* We tried another kind of product of designs $D_1 = (V_1, E_1)$ and $D_2 = (V_2, E_2)$, with again $V_1 \times V_2$ as vertex set, but with one edge $V_1 \times e_2$ for each edge $e_2 \in E_2$ and one edge $e_1 \times V_2$ for each $e_1 \in E_1$. This "full" product was applied to pairs of complete graphs.

*vi.* Last, we considered "tensors", generalizations of matrix designs where we take successive Cartesian products of single-pool designs : $\mathrm{SP}_n^* \,\square\, \mathrm{SP}_n^* \,\square \ldots \square\, \mathrm{SP}_n^* = \left(\mathrm{SP}_n^*\right)^{\square d}$ where the square in exponent means that we take a power with respect to the Cartesian product. They can be described as higher-dimensional "arrays", each sample being identified by a $d$-vector of coordinates with values in $\{1, \ldots, n\}$, and pools correspond to "cardinal lines" where all samples differ by only one coordinate. $\left(\mathrm{SP}_n^*\right)^{\square d}$ has $N = n^d$ samples, each in $L = d$ pools, and $M = dn^{d-1}$ pools of size $K = n$. Since we can freely tune $L$ and $K$ and since pools intersect at at most one one samples, Theorem 3 of [MT11] applies: by choosing $n$ and $d$ adequatly in term of $p$, $\mathrm{ETS}(\left(\mathrm{SP}_n^*\right)^{\square d}, p)$ can be made minimal among all designs within a multiplicative constant tending to 1 as $p$ goes to 0. This gives explicit examples of asymptotically optimal designs, instead of the random designs proposed in [MT11]. However, tensors are absurdly large when $n$ and $d$ increase (e.g. $n = 10$ and $d = 6$ already yields a million samples), and the few ones that satisfy constraints (1) and (2) did not perform as well as some other designs.

### 6.8 Number of second-round tests

A close inspection of the curves of $p \mapsto \mathrm{ETS}(p, D)$ shows an interesting phenomenon: for $D = \mathrm{SP}_n$ this function has positive derivative at 0, so that the number of second-round tests is roughly proportional to $p$; but when $D$ is 1-perfect, the derivative is 0 at 0, see Figure 7 (this is also easily proven from the definition of derivative). Having vanishing derivative at 0 means that for small $p$ the number of second-round test is much smaller than proportional in $p$. In fact, the optimality range of the designs in Table 1 can be observed to correspond to the region where the derivative in $p$ of $\mathrm{ETS}(p, D)$ starts to increase steeply, and this is easily understood: for smaller $p$, the number of first-round tests is higher than for other designs, and for larger $p$ the number of second-round tests increases – this phenomenon is typical when considering the envelope of a family of curves.

To make this remark quantitative, let us extract from Table 1 the average number of second-round test per sample for a selection of values of $p$. This is obtained simply as $\mathrm{ETS}(p, D) - \frac{M}{N}$, since $\mathrm{ETS}(p, D)$ is the average total number of tests, and $\frac{M}{N}$ the number of first-round tests. The values are given in Table 2. We observe that at $p = 13\%$, while the single-pool-of-three is as efficient as $K_6^*$ in terms of ETS, it induces significantly more second-round tests. Both the number of second-round tests and their proportion among performed tests decrease rapidly with $p$, when 1-perfect designs are used. At $p = .1\%$, e.g. when broadly screening a mildly affected population, the design $K_{44}^*$ induces a twentyfold decrease in the overall number of tests needed, while only incuring second-round-induced delay for .2% of the individual tested. This show that two-round testing can be both efficient and quick. For very small $p$, while single-pool designs with large $n$ are able to fit into the constraint (2), they induce a large number of second-round tests.

## 7 Heuristics for searching good designs.

In this closing section we discuss the search for efficient pooling designs, starting with heuristics provided by information theory.

### 7.1 Information theory and efficient designs

The basic tool is the *entropy function*: if $X$ is a random variable taking $n$ values $x_1, \ldots, x_n$ with probabilities $p_1, \ldots, p_n$ (so that $p_i > 0$ and $\sum_i p_i = 1$), one defines

$$H(X) = H(p_1, \ldots, p_n) = \sum_i p_i \log_2 \frac{1}{p_i}.$$

The main properties of $H$ are:

*i.* if a random variable $Y$ can be expressed as a function of $X$, then $H(Y) \leq H(X)$,

*ii.* if $X$ takes only two values, then $H(X) \leq 1$ with equality when these values are taken with probability $\frac{1}{2}$,

*iii.* if $X_1, \ldots, X_k$ are random variables, we can consider the aggregate random variable $X = (X_1, \ldots, X_k)$ giving the tuple of their outcomes; then $H(X_1, \ldots, X_k) \leq \sum_j H(X_j)$ with equality when the $(X_j)$ are independent.

When performing two-rounds group testing, we reconstruct the random variable $Y$ giving the status of our $N$ individual, positive or negative, from the random variable $X$ aggregating the results of both rounds. That this reconstruction is possible implies that $Y$ can be expressed as a function of $X$, so that $H(X) \geq H(Y) = Np \log_2 \frac{1}{p}$ when the individual are positive with probability $p$ and independent from each other, as in our model. Meanwhile, $X$ consists in a random number of

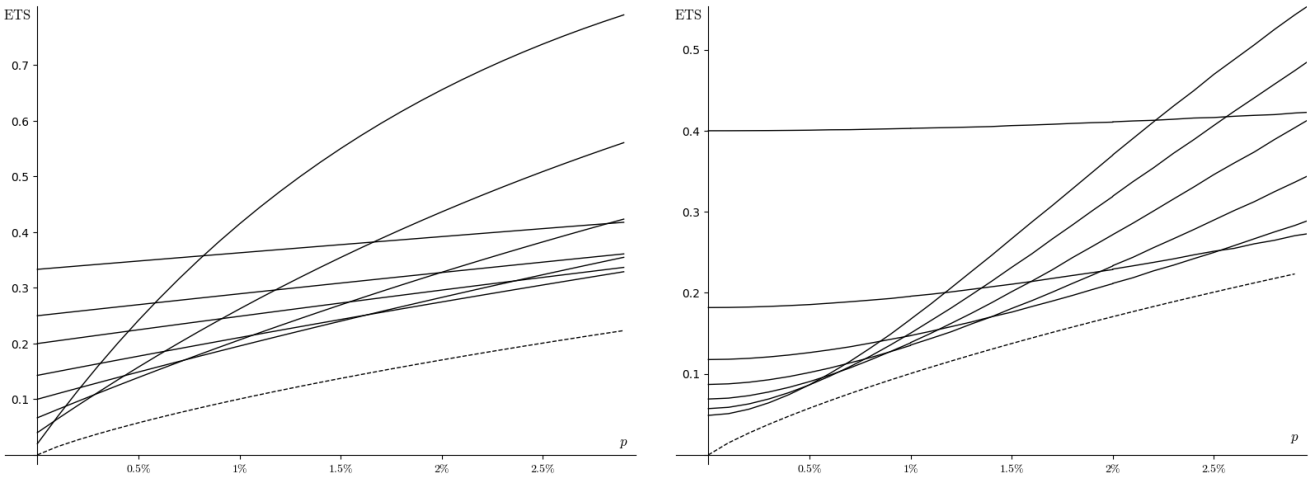


Figure 7: The curves $p \mapsto \mathrm{ETS}(p, D)$ have positive derivatives at 0 for $D$ a single-pool design (left, selection of $\mathrm{SP}_n$), but vanishing derivative at 0 for 1-perfect designs (right, selection of $K_n^*$)

| Design | $p$ | ETS | Second-round tests per sample |
|---|---|---|---|
| $\mathrm{SP}_3$ | 20% | .82 | .49 |
| | 13% | .67 | .34 |
| $K_6^*$ | 13% | .67 | .27 |
| $K_8^*$ | 8% | .50 | .21 |
| $\mathrm{STS}_{25}^*$ | 5% | .36 | .11 |
| $\mathrm{STS}_{33}^*$ | 3% | .26 | .07 |
| $\text{4-BIBD}_{85}^*$ | 2% | .19 | .04 |
| $\mathrm{STS}_{67}^*$ | 1% | .12 | .03 |
| $K_7^* \boxtimes K_{10}^*$ | .5% | .084 | .010 |
| $K_{44}^*$ | .1% | .049 | .002 |
| $\mathrm{SP}_{100}$ | $\to 0$ | $\to .01$ | $\sim 100p$ |

Table 2: *Average number of second-round tests per sample for optimal designs.*

bits of information, on average $N\,\mathrm{ETS}(D,p)$. It follows that $H(X) \leq N\,\mathrm{ETS}(D,p)$, implying

$$\mathrm{ETS}(D,p) \geq p \log_2 \frac{1}{p}$$

which is the *information bound* we mentioned. This bound can be strengthened, at least for low $p$, for two-round pooled testing [MT11]. Trying to get as close to the information bound as possible and taking inspiration from the proof of the improved bound, we can give good heuristics for important parameters of designs.

First, let us look at the first round. Using a greedy approach, we can try to have every test provide as much information as possible, meaning we would like

*i.* the probability for each pool to be positive to be as close to $\frac{1}{2}$ as possible, and

*ii.* the results of different pools to be as independent as possible.

The first condition is realized by pools all of the same size $K$, where $K$ should make $(1-p)^K$ as close to $\frac{1}{2}$ as possible. Values are shown in Table 3, where we compare with the designs that performed best in our experiments. Some discrepancy is expected due to the crudeness of the heuristic; at very small $p$, the constraints we put on the design seem to have a strong effect toward smaller pools.

The second condition can be fully realized only by taking disjoint pools; but this prevents 1-perfectness of the design, and more importantly it will result in too much loss of information in the second round. Some intersection being necessary, keeping the intersections frequently small will get us close to independence. In particular, asking every two pools to intersect in at most one sample is a good way to ensure almost-independence of pool results.

Let us turn to the second round. It could feel irrelevant as the number of tests performed then is directly decided by the results of the first round; but the information bound tells us that this number must be large enough to cover the information missed in the first round. It can only be low if each second-round test provides a large amount of information, which we can try heuristically to optimize. Again, we would like each test to be positive with a probability close to $\frac{1}{2}$, and all second-round tests to be as independent as possible. For independence, we want again pools to intersect as little as possible, so that different sample of unknown status are not linked too strongly; we are again led to assume that any two different pools intersect in at most one sample. To estimate the information provided by testing an unknown sample, by a slight simplification we can consider the probability of being tested positive for a sample $x$ of unknown status *knowing* that the $L$ pools containing it are positive. This disregards some information, but likely a negligible amount at least for small enough $p$ (see again [MT11]). With the assumption just made, and assuming all pools have the same size $K$, this conditional probability is

$$\frac{p}{p + (1-p)\left(1-(1-p)^{K-1}\right)^L}.$$

Given $K$ (e.g. the heuristically optimal one for the first round), we can thus optimize $L$ to make this as close as $\frac{1}{2}$ as possible. For example, we observe that at $p = 30\%$, taking $K = 2$ or $K = 3$ leads to $L = 1$, but $K = 3$ gives a result closer to $\frac{1}{2}$. This can be seen as an explanation for the design with a single pool of 3 performing better than the design with a single pool of 2. However in general it is difficult to define an optimal pair $(K, L)$, since we do not know the relative weight of the first and second round.

Table 3 compares at different values of $p$ the values of $K$ and $L$ suggested by this heuristic with their values for the best designs we tested. At low $p$, the size of pools is brought down in practice by the constraint on the size of designs; observe that using a tensor product, we can easily realize a 1-perfect design with $K = 99$ and $L = 7$, but with exceedingly large size $N = 99^7 > 9 \cdot 10^{13}$. We also see some discrepancy early on in the number of pools each sample should be in, which is higher in the heuristic than in the best design we know of. It rather seems to be a limitations of the heuristic reasoning since for example for $p = 11\%$, we did test a design with $K = 6$ and $L = 3$ (a tensor design) but it performed significantly worse than duals of complete graphs.

## 7.2 The role of constraint (2) at small $p$

To better understand why the $K_n^*$ perform so well at small $p$, let us explain how, for very small $p$, the constraint (2) kicks in to strongly restrain the available designs.

**Theorem 1.** *Let $D = (V, E)$ be a 1-perfect design where any two pools intersect in at most one sample, and satisfying constraint (2): $N \leq 1000$. Then its number of pools $M$ satisfies*

$$\frac{M}{N} \geq \frac{1 + \sqrt{8001}}{2000} > 0.045224.$$

Recall that $M/N$ is the limit of $\mathrm{ETS}(D,p)$ as $p$ goes to 0; and $K_{45}^*$ has $N' = 990$ samples and $M' = 45$ pools, thus

$$\lim_{p\to 0} \mathrm{ETS}(D,p) > .995 \times \frac{45}{990} = .995 \lim_{p\to 0} \mathrm{ETS}(K_{45}^*, p),$$

i.e. for small enough $p$, under the three constraints of the above theorem, $K_{45}^*$ cannot be improved by more than half a percent. At very small $p$, one is thus lead to either give up on 1-perfectness (but then any increase in $p$ will increase sharply ETS and delay many tests to the second-round); or stick with $K_{45}^*$. The most promising direction to improve on this current state of affairs seems to seek 1-perfect designs with close to 1000 samples and less than 40 pools, intersecting pairwise in 2 samples.

| $p$ | heuristic suggested | | Best known designs |
|---|---|---|---|
| | $K$ | then $L$ | $(K, L)$ |
| 25% – 30% | 2 | 1 | $(3,1)$ |
| 18% – 24% | 3 | 1 | $(3,1)$ |
| 15% – 17% | 4 | 2 | $(3,1)$, $(5,2)$ |
| 12% – 14% | 5 | 2 | $(3,1)$, $(5,2)$, $(6,2)$, $(7,2)$ |
| 11% | 6 | 3 | $(5,2)$, $(6,2)$, $(7,2)$ |
| 9% – 10% | 7 | 3 | $(5,2)$, $(6,2)$, $(7,2)$, $(8,2)$ |
| 8% | 8 | 3 | $(6,2)$, $(7,2)$, $(8,2)$ |
| 7% | 10 | 4 | $(7,2)$, $(8,2)$ |
| 6% | 11 | 4 | $(8,2)$, $(12,3)$, $(13,3)$ |
| 5% | 14 | 4 | $(12,3)$, $(13,3)$ |
| 4% | 17 | 4 | $(13,3)$, $(16,3)$ |
| 3% | 23 | 5 | $(16,3)$, $(18,3)$, $(19,3)$, $(24,4)$, $(25,4)$ |
| 2% | 34 | 5 | $(28,4)$, $(29,4)$, $(32,4)$, $(33,4)$ |
| 1% | 69 | 7 | $(33,3)$, $(36,3)$ |
| .7% | 99 | 7 | $(36,3)$, $(54,4)$ |
| $\leq .6\%$ | $> 100$ | | |

Table 3: *Heuristic optimization of parameters.*
Parameter suggested by the information heuristic, without constraints (1) and (2), are compared with the experimentally best performing designs (which abide by constraints (1) and (2)).

*Proof.* As explained in Section 5.3.6, we can assume that for each sample $s \in V$, there are at least two elements in the set $E_s$ of all pools containing $s$ (otherwise either $D$ is not 1-perfect, or there is a pool with a single sample which can be removed while improving $D$).

In the bipartite graph representation of $D$, we consider the set

$$A = \{(s, \{g_1, g_2\}) \mid s \in V, g_1 \neq g_2 \in E, s \in g_1, s \in g_2\}$$

of pairs of incidences $(s \in g)$ that share the same sample. We count the number of elements of $A$ in two different ways. First, every $s$ has at least one incident pair $\{g_1, g_2\}$ so that $|A| \geq N$. Second, since pairs of pools intersect in at most one sample,

$$\begin{aligned} |A| &= \sum_{\{g_1 \neq g_2 \in E\}} \big|\{s \in V \mid s \in g_1, s \in g_2\}\big| \\ &\leq \big|\{\{g_1 \neq g_2 \in E\}\big| = \frac{M(M-1)}{2}. \end{aligned}$$

Denoting by $\alpha$ the ratio $M/N$, combining both counts we get $N \leq \alpha N(\alpha N - 1)/2$. Simplifying and using $N \leq 1000$ yields

$$2 \leq \alpha(1000\alpha - 1).$$

This is a degree two inequality, developing and completing the square we obtain $\alpha \geq (1 + \sqrt{8001})/2000$. □

A similar computation shows that if we further restrict to regular designs, to beat $K^*_{45}$ at small $p$ it is necessary that $L = 2$; and if we wanted to take $L \geq 3$ and relax (2), we would have $N$ significantly larger than 2000.