




Optimal multi-stage group partition for efficient coronavirus screening

Yuvraj Gajpal¹ · S. S. Appadoo¹ · Victor Shi² · Guoping Hu³ 

Accepted: 18 January 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

The outbreak of COVID-19 has affected the economy worldwide due to entire countries being on lockdown. This has been highly challenging for governments facing constraints in terms of time and resources related to the availability of testing kits for the virus. This paper develops an optimal method for multiple-stage group partition for coronavirus screening using a dynamic programming approach. That is, in each stage, a group of people is divided into a certain number of subgroups, each will be tested as a whole. Only the subgroup(s) tested positive will be further divided into smaller subgroups in the next stage or individuals at the last stage. Our multiple-stage group partition scheme is able to minimize the total number of test kits and the number of stages. Our scheme can help solve the test kit shortage problem and save time. Finally, numerical examples with useful managerial insights for further investigation are presented. The results confirm the advantages of the multi-stage sampling method over the existing binary tree method.

Keywords COVID-19 · Pandemics · Group testing · Multi-stage group partition · Optimization · Dynamic programming

✉ Guoping Hu
hugp@swufe.edu.cn

Yuvraj Gajpal
yuvraj.gajpal@umanitoba.ca

S. S. Appadoo
SS.Appadoo@umanitoba.ca

Victor Shi
cshi@wlu.ca

¹ Asper School of Business, University of Manitoba, Winnipeg, MB R3T 5V4, Canada

² Lazaridis School of Business and Economics, Wilfrid Laurier University, Waterloo, ON N2L 3C5, Canada

³ School of Business Administration, Southwestern University of Finance and Economics, Chengdu, China

1 Introduction

The World Health Organization (WHO) declared a pandemic over a new coronavirus on March 11th, 2020 after observing the spread of the virus over 114 countries with 118,000 cases and 4291 deaths (WHO, 2020). The spread of the new coronavirus has globally affected the supply of goods and materials (Zandkarimkhani et al., 2020; Ivanov, 2020; Queiroz et al., 2020). The supply chain disruption due to COVID 19 is causing economic downfall worldwide, which is evident from the negative GDP growth rate in several countries around the world (Saltzman (2020)). The supply chain disruption is expected to persist till herd immunity is achieved; till then, every country will rely on vaccination, and testing and isolating coronavirus patients to minimize the spread of diseases.

An effective strategy to fight against the coronavirus is to aggressively test as many people as possible. More testing helps to know the movement of the virus through the population, tracing of the infected people, isolating their contacts, and identifying the hot spots for locking down, all of which lead to the prevention of large-scale spread. However, almost every country is suffering from the scarcity of testing kits to accurately track the new coronavirus. To help deal with the scarcity of testing kits, many news and research articles advocate the group pool testing method for coronavirus screening (Biswas, 2020; Conger, 2020; Cosh, 2020; Gossner & Gollier, 2020; Eberhardt et al., 2020, Sinnott-Armstrong, (2020), Yelin et al., 2020, Brault et al., 2021). The group pool testing method allows ruling out a considerable number of samples with one test. According to Cosh (2020), Technion, Israel Institute of Technology has conducted a successful trial of group pooling testing method using polymerase chain reaction (PCR) for the presence of the coronavirus. The Group pool testing method mixes multiple swabs and performs a single test for the whole group. If the group test result is negative, then the whole group is ruled out for virus infection. Otherwise, further testing of each individual is performed in the infected group to identify the infected people.

This paper exhibits a situation of group testing where a positive case is present. According to Cosh (2020), the pooled test with 64 people at a time could yield a positive test result even if only one person in the group is infected with the coronavirus. Thus, the remainder of the paper assumes that a group pool testing of 64 people is possible with reasonable accuracy. This paper finds an effective way to identify the person infected with coronavirus in a group of 64 samples, which has shown a positive report. We also propose and demonstrate a generalized method for identifying infected persons among n samples.

The basic group pool testing approach states that if a group pooling test of 64 samples is found to be positive, then further testing of individuals in the group is performed to identify a person infected with the coronavirus. This testing method requires a total of 65 test kits to identify the infected person. The basic group testing requires two times swab collection from 64 individuals. The first sample is collected for the group pool testing, and the second sample is collected for the individual testing. Without loss of generality, we assume that the swab collection is time-sensitive, and thus, multiple swab collections for later use can affect the testing accuracy. If the 64 samples can be further divided into different subgroups, then the number of required test kits can be further reduced. Although the subgroup testing increases the number of swab collections, the additional cost can be used as a part of a trade-off with the reduction of the required total number of test kits. The existing multi-stage group testing method uses a binary tree method to identify the infected person. In the binary tree method, a group with a positive report is further divided into two groups in the next stage. The existing binary tree method is shown in Fig. 1 for a 16 sample.

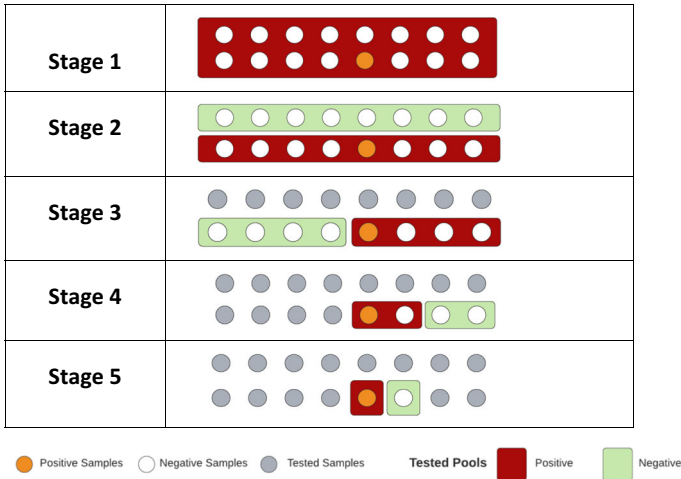


Fig. 1 A binary tree group testing method for 16 sample

A disadvantage of the existing multi-stage method is that it does not consider the minimization of stages. To overcome this problem, the current paper proposes a general multi-stage group testing method in which a group with a positive report can be divided into any number of groups in the next stage. The proposed multi-stage sampling method finds the optimal regrouping strategy in different stages. The primary objective is to minimize the total number of test kits, and the secondary objective is to minimize the number of stages.

This paper contributes to the literature in many ways. This paper introduces a deterministic model as opposed to the probabilistic model in the group testing method. The group testing model found in the literature considers the fixed number of subgroups at every stage. This paper introduces a generalized version of the multi-stage group testing methods in which the number of subgroups varies in each stage. In our knowledge, the dynamic programming (DP) approach is the first time used in this paper to analyze the group testing methods. The dynamic programming approach finds the optimal number of test kits for the group testing method. This paper also illustrates how a basic optimization model can address medical/clinical/healthcare questions.

The remainder of the paper is organized as follows. Section 2 provides a basic background of the stage sampling method with a numerical example. Section 3 briefly reviews the literature on group pool testing and dynamic programming. Section 4 provides a general problem definition for n samples and develops a dynamic programming method to solve the proposed multi-stage problem. Section 5 provides numerical experiments and managerial insights. Lastly, the paper is concluded in Sect. 6.

2 Problem background and multi-stage sampling method

This paper defines the term *stage sampling method* in which the stage represents the maximum number of swab samples collected to identify the infected person. The *one-stage-sampling-method* requires just one swab sample collection; the *two-stage-sampling-method* requires two swab collections etc. Most countries are currently using the *one-stage sampling method*

in which just one swab sample per person is collected to identify the infected person. The basic group testing method falls under the category of *two-stage-sampling-method* because, at maximum, it requires two swab sample collections per person. The first swab sample is collected for group testing at the first stage, and the second swab sample is collected for individual testing at the second stage.

We explain the *three-stage-sampling-method* using a numerical example. The example seeks to find an infected person among 64 individuals. The objective is to minimize the number of test kits. We assume that only one person is infected out of the 64 individuals. In the *three-stage-sampling-method*, three swab samples are collected to identify the person infected with the coronavirus. First, a group test of 64 samples is performed at the first stage. In the second stage, 64 individuals are further divided into m sub-groups, and each sub-group is tested. Since there is only one person infected among 64 individuals, only one of the subgroups would show a positive report, and the remaining $(m-1)$ subgroups would show a negative report. The members from the subgroup with the positive report are finally tested individually at the third level. In the *three-stage-sampling-method*, the number of feasible solutions depends on the number of subgroups formation at the second stage. Some of the feasible solutions are shown in Table 1. We do not list all the possible feasible solutions because, after some point, the objective function starts increasing due to the convex property of the problem.

We illustrate the calculation of objective function (i.e., number of test kits) for solution 2. In solution 2, the first 64 swab samples are tested at the first level. At the second level, swab samples are collected for the second time for all 64 individuals, but they are divided into three subgroups with 21, 21, and 22 samples, respectively. Finally, at the third level, all 22 individuals will be tested in the worst-case scenario. Solution 2 requires 1 test kit at the first stage, 3 test kits at the second stage, and 22 test kits at the third stage. Therefore, a total of 26 test kits is required under solution 2.

It is clear from Table 1 that when the number of subgroups increases at the second level, the number of individual tests at the third level reduces, which eventually reduces the total

Table 1 A numerical example of the three-stage sampling method

Feasible solution	Number of subgroups at the second stage	Size of group samples		Individual samples stage 3	Total number of test kits
		Stage 1	Stage 2		
Solution 1	2	64	32, 32	32	35
Solution 2	3	64	21, 21, 22	22	26
Solution 3	4	64	16, 16, 16, 16	16	21
Solution 4	5	64	12,13,13,13,13	13	19
Solution 5	6	64	10,10,11,11,11,11	11	18
Solution 6	7	64	9,10,10,10,10,10,10	10	18
Solution 7	8	64	8,8,8,8,8,8,8	8	17
Solution 8	9	64	7,7,7,7,7,7,7,8	8	18
Solution 9	10	64	6,6,6,6,6,6,7,7,7	7	18
Solution 10	11	64	5,5,6,6,6,6,6,6,6,6	6	18
Solution 11	12	64	5,5,5,5,5,5,5,5,6,6,6,6	6	19

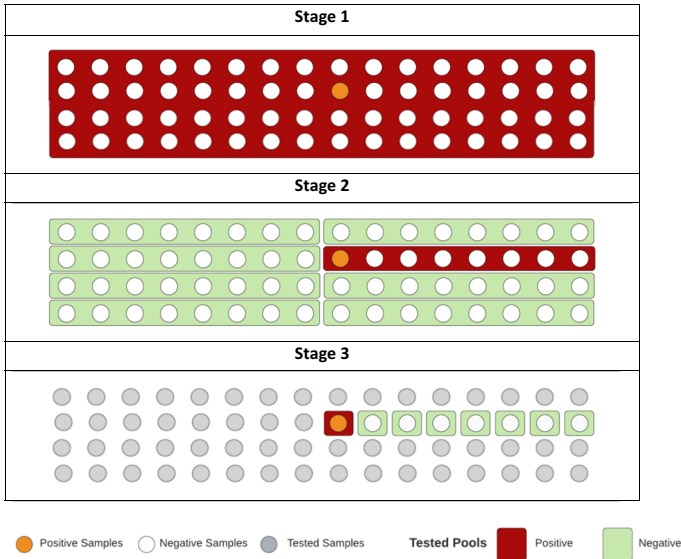


Fig. 2 Optimal group partition for 64 samples in three stages

number of required test kits. However, after some point, an increasing number of subgroups does not bring any benefit in reducing the total number of required test kits. That is to say, the total number of test kits required increases after some optimal level of subgrouping. If the number of test kits is considered as a function of the number of groups at the second level, then this function is convex. This property can be verified using the recursive equation described in Sect. 4.

As shown in Table 1, the optimal solution (i.e., solution 7) has an objective function value of 17. In solution 7, 64 samples are tested in the group at the first stage, and then these 64 individuals are divided into 8 groups with 8 members in each group. Finally, at the third level, 8 samples are tested individually. This sub-grouping method is shown in Fig. 2.

3 Literature review

The group testing problem has received considerable attention recently after the coronavirus epidemic (e.g., Ben-Ami et al., 2020, Delaigle et al., 2020, Khodare et al., 2020). The popularity of the group testing can be seen from the number of articles available in medRxiv archive, which provides an archive for unpublished manuscripts in the medical, clinical, and related health science (<https://www.medrxiv.org/>). The research papers available in this archive advocate the use of group testing for efficient screening of coronavirus (Yelin et al., 2020, Sinnott-Armstrong, 2020). The general group testing problem aims to identify a small set of k infected individuals out of population size (Aldridge et al., 2014, Aldridge, 2018). In Aldridge et al. (2014), they focused on the group testing problem where the test structure is given via a Bernoulli random process. They then compared the performance of four detection algorithms via simulations. In Aldridge (2018), they studied nonadaptive probabilistic group testing in the linear regime, where each item is defective independently.

The group testing protocol was first introduced by Dorfman (1943) during World War II to efficiently remove the Army men who had syphilis. The work of Dorfman (1943) was further advanced in the literature, such as Saraniti (2006), Feng et al. (2010), and Li et al. (2014). All these methods assumed that the test results were perfect. Specifically, Saraniti (2006) proposed three pooled testing techniques for generic testing conditions and then applied them to demonstrate potential costs savings for universal HIV screening in the United States and Thailand. Feng et al. (2010) developed a general two-stage model that uses stochastic dynamic programming at stage 2 for the optimal group sizes and nonlinear programming at stage 1 for the optimal number of group-testable units. Li et al. (2014) developed group both adaptive and non-adaptive testing algorithms to identify with high probability the subset of defectives via nonlinear (disjunctive) binary measurements.

More recently, Aprahamian et al. (2019) studied the group testing method for subject-specific risk characteristics and imperfect tests. They reduced the problem as a partition problem and developed efficient algorithms to solve the problem.

Collier and Gossner (2020) advocated the group testing method as an efficient strategy for identifying the presence of the virus. Specifically, they proposed a testing protocol to identify an infected person by dividing a positively reported group into two subgroups for testing in the next stage.

Assad et al. (2020) relied on numerical simulations to check the effectiveness of the group testing protocol using the Binary Elimination Algorithm. They find that the total number of test kits depends on the pool size and the percentage of positive samples, i.e., the percentage of virus carries in the population. In particular, the group testing protocol can be ineffective for reducing the number of required test kits if the percentage of positive samples is reasonably large (e.g., 4%).

The method proposed by Collier and Gossner (2020) and Assad et al. (2020) can be named a Binary Tree Method because it divides the group into two groups in every stage. Eberhardt et al. (2020) developed a multi-stage testing scheme that tests samples in groups of various pool sizes in multiple stages. This scheme is designed based on two integers: y (divisor) and k (number of stages). The initial pool size $N = y^{k-1}$ is divided by y into each subsequent stage, resulting in pool sizes $y^{k-1}, y^{k-2}, \dots, y^0$, where $y^0 = 1$. By setting $y = 2$, the multi-stage group testing method reduces to the binary tree method.

The method proposed in this paper is a generalized version of the multi-stage group testing method, which allows different groups in different stages. Furthermore, while the group testing methods available in the literature do not consider the minimization of the number of stages, our model considers the minimization of the number of stages along with the minimization of the number of tests. In addition, our method provides an optimal solution for a specified number of stages as well. The optimization in our model is implemented using dynamic programming.

The group partition problem considered in this paper is closely related to the cluster analysis problem in which the group is partitioned optimally. The application of a dynamic programming approach for solving cluster analysis problems can be found as early as Jansen (1969). The clustering problem is also known as the k -clustering problem because it involves partitioning n entities into k disjoint and nonempty subsets such that some objective function can be optimized. The recursive equation proposed by Jansen (1969) was improved by Van et al. (2004) by removing the duplicate calculation. They proposed redundant and quasi-non redundant methods to solve the clustering problem. Jessop (2010) considered a variant of the k -clustering approach, which determines the number of clusters along with the members of each cluster. Brusco et al. (2017) considered another variant of the k -clustering problem based on Gaussian model-based partitioning criteria. Babaki et al. (2014) considered a constrained

clustering problem, where the constraints considered are must link constraints, can not link constraints and anti-monotone constraints on individual clusters. Aloise et al. (2012) proposed column generation embedded with branch and bound to solve the problem. Our problem is different from the k -clustering problem in the way that the objective function of the cluster is equal to the size of the cluster. Also, our problem partitions a group into many stages, and the grouping in a given stage depends on the grouping at the immediately preceding stage.

A dynamic programming approach has been used to solve various problems. The popularity of dynamic programming can be seen from the fact that Google Scholar's search with the keyword "dynamic programming" produced more than 100 articles in the first 4 months of the year 2020. Some examples of dynamic programming applications are lameness detection in dairy herds (Kanyamattam et al., 2020); reservoir operation (Rani et al., 2020); peak detection in Genome data (Hocking et al. (2020)); power optimization (Xu et al., 2020); periodic review model (Voelkel et al., 2020); aircraft maintenance scheduling problem (Deng et al., 2020); and aeromedical evacuation dispatching (Robbins et al., 2020, Summers et al., 2020). These studies confirm that the dynamic programming approach indeed has a wide variety of applications.

4 Problem definition and dynamic programming approach

This section first presents the description of the multi-stage group testing (MSGT) problem, and then it describes the dynamic programming approach to the MSGT problem with a known number of stages.

4.1 Problem definition

The multi-stage group testing (MSGT) problem for coronavirus screening can be defined as follows. There is a group of n people, and only one of them is the virus carrier. This group can be tested either in a group or individually. Each test provides either a positive or a negative report. If a group test is found to be negative, then every member of the group is considered a negative. If a group test is found to be positive, then further testing is performed in the next stage for the group. The complete test is performed in T stages. In the first stage, a single group test of n samples is performed. From stage 2 to stage $(T-1)$, group tests are performed for more than one sample at each stage. In the second stage, n samples are divided into the m_2 groups. Similarly, in other stages, the required samples are divided into different groups. Finally, at the T^{th} stage, an individual test is performed. Each group test and each individual test require just one medical kit. The problem involves finding the optimal number of subgroups for each stage from stage 2 to stage $(T-1)$ (i.e., $m_2^*, m_3^*, \dots, m_{T-1}^*$) to identify the member infected with the coronavirus. The first stage tests n samples in a group, while the last stage tests remaining samples individually. Therefore, the first and the last stage do not require finding optimal groups. The group partitioning problem for coronavirus testing aims to minimize the number of medical kits of n samples in T stages.

4.2 Dynamic programming approach

The MSGT problem can be optimally solved using dynamic programming (DP) approach. The proposed DP approach mainly solves T -stage problem to find out the minimum number of required test kits. The dynamic programming (DP) problem needs recursive equations

to identify the minimum number of required test kits at t th stage (Winston, 2004). The DP requires the definition of ‘stage’ and ‘state’. The ‘stage’ defined in the problem represents the ‘stage’ of DP. The number of group formation in the immediate next stage represents the ‘state’. We define the following terminologies to identify positive reports among n mixed samples in T stages:

$F_t^{m_{t+1}}(l)$ The number of optimal required kits for l samples in remaining $T - t$ stages, if m_{t+1} group tests are performed in the $(t + 1)^{th}$ stage.

$F_t(l)$ The number of optimal required kits for processing l samples in remaining $T - t$ stages.

f^m The number of required kits at the $(t + 1)^{th}$ stage for testing m groups.

The proposed DP approach works in a backward direction. First, we find the optimal solution for $(T - 1)^{th}$ stage and then move back towards stage 1.

4.2.1 Initialization and boundary condition

Our DP approach starts with the $(T - 1)^{th}$ stage which is the last stage when group test is performed. The last stage is the T^{th} stage in which an individual test is performed. The optimal number of test kits required at the $(T - 1)^{th}$ stage can be initialized as follows.

$$F_{T-1}(l) = 1 + l \quad \forall l = 2, \dots, n \quad (1)$$

In this equation, one group test is performed at $(T - 1)^{th}$ stage and then l individual test is performed at the last stage (i.e., T^{th} stage). Hence, the total of $1 + l$ test kits is required if the second last stage performs a group test of l samples. The boundary condition for one sample at t th stage can be initialized as follows:

$$F_t(1) = \infty \quad \forall t = 1, \dots, T - 1 \quad (2)$$

The above equation indicates that a group test with only sample is not allowed at any stage from 1st till $(T - 1)^{th}$ stage.

4.2.2 Recursive equations of remaining stages

After initializing the $(T - 1)^{th}$ stage, the DP programming finds optimal solution for remaining $T-2$ stages, starting from $(T - 2)^{th}$ stage to the 1st stage. The following recursive equation is used to calculate $F_t^{m_{t+1}}(l)$:

$$F_t^{m_{t+1}}(l) = f^{m_{t+1}} + F_{t+1}(\lceil l/m \rceil) \quad \forall m = 2, \dots, l \quad (3)$$

In this equation, the first term represents the number of required test kits at the immediate next stage. If m_{t+1} group tests are performed at the $(t + 1)^{th}$ stage, m_{t+1} test kits are required (i.e., $f^{m_{t+1}} = m_{t+1}$). The second term represents the optimal kits needed if $\lceil n/m \rceil$ requires further testing to identify the infected person in remaining $T - t$ stages. The term $\lceil n/m \rceil$ represents the nearest integer for l/m rounded up.

In the $(t + 1)^{th}$ stage, the optimal m_{t+1} groups are formed from l samples. The optimal m_{t+1} partition aims to minimize the number of samples with the largest group. Simple observation shows that the size of the largest groups will be $\lceil l/m_{t+1} \rceil$ in the optimal partitioning of l samples in m_{t+1} groups. Consider a simple example of partitioning 13 samples into 5 groups. In the optimal partitioning, 2 samples will be assigned to each group initially, which will make a total 10 sample assignments to different groups. The remaining 3 samples can be assigned to any three groups, with 1 sample in each group. Thus, the optimal 5 partitioning

for 13 samples will be 2,2,3,3 and 3. In the worst-case scenario, the largest group will show a positive report. In this case, further testing is required for the largest group with $\lceil l/m \rceil$ members in the remaining $T - t$ stages. The second term of Eq. (1) represents the optimal solution for testing $\lceil l/m \rceil$ swab samples in $t-1$ stages. Given the definition of $F_t^{m_{t+1}}(l)$, the optimal value of $F_t(l)$ can be expressed as follows:

$$F_t(l) = \min_{m_{t+1}=2, \dots, l/2+1} (F_t^{m_{t+1}}(l)) \quad \forall t = 1, \dots, T - 3 \tag{4}$$

The value $F_1(n)$ provides the optimal number of test kits for processing n samples in T stages. The primary objective of the MGMT problem is to minimize the required test kits for a given value of T , while the secondary objective is to minimize the number of stages. The DP approach described in this section is mainly designed for the primary objective. The problem can be repetitively solved for different values of T to achieve the secondary objective.

5 Numerical results

The dynamic programming method proposed in this paper is a general approach with a few lines of code in any language. We use C++ language to code the proposed DP problem. We provide numerical experiments for 64 samples in Sect. 5.1. We then illustrate the optimal partition for 64 samples and discuss the findings. The results for 64 samples are useful for a country like Israel which has developed the method for testing up to 64 samples with accuracy. In Sect. 5.2, we provide the optimal solution for sample sizes starting from 5 up to 100. The results provided there can be used by any country based on their capability to process group samples with accuracy. In Sect. 5.3, we provide managerial insights into the proposed group testing method.

5.1 Numerical experiments for a sample size of 64

This subsection describes the optimal partition of 64 samples in T stages. The optimal solution for a 3-stage sampling method for 64 samples is already illustrated in the problem background. The summary of the optimal solution for the 3-stage group testing problem until the 7-stage group testing problem is summarized in Table 2. We don't need a group testing method with more than 7 stages because dividing 64 samples into more than 7 stages brings 1 sample

Table 2 The optimal group partition for 64 samples

Number of Stages (T)	Size of group testing ()							Min number of kits
	1st Stage	2nd Stage	3rd Stage	4th Stage	5th Stage	6th Stage	7th Stage	
3	64	8,8,8,8,8,8,8,8	8*					17
4	64	16,16,16,16	4,4,4,4	4*				13
5	64	32,32	16,16	4,4,4,4	4*			13
6	64	32,32	16,16	8,8	4,4	4*		13
7	64	32,32	16,16	8,8	4,4	2,2	2*	13

* Indicates individual tests

test in some of the earlier stages, which increases the total number of kits requirements. Our dynamic programming model provides a *BigM* value for more than 7 stages because of the boundary condition shown in Eq. (4).

The results reported in Table 2 show that the 3-stage group testing method needs 17 kits. The number of kits reduces to 13 for the 4-stage group testing method. The total number of required kits does not decrease further for an increasing number of stages. The results bring an interesting insight into the group partitioning problem. For a sample size of 64, the binary testing method used in current literature seems to provide the optimal solution (in terms of the number of kits) for the multi-stage group testing problem. In Table 2, the binary testing method corresponds to the 7-stage group sampling method. In the 7-stage group testing problem, groups are divided into two groups in every stage. The 7-stage group testing problem requires 13 kits which is the minimum number of kits requirement. However, the minimum number of kits requirements can be obtained at the 4-stage group sampling problem as well as illustrated in Fig. 3. In the 4-stage sampling problem, the first 64 samples are tested in the 1st stage. In the second stage, 64 samples are divided into 4 groups with 16 members in each group. One of these groups would show a positive report, and thus 16 members have divided again in the 3rd stage. In the 3rd stage, 16 members are divided into 4 groups with 4 members in each group. One of these groups will show a positive result which is finally tested individually at the 4th stage. The 4-stage sampling requires 1 kit at the first stage, 4 kits at the second, third, and fourth stage. Thus, for the optimal 4-stage group sampling, the total number of kits required is 13.

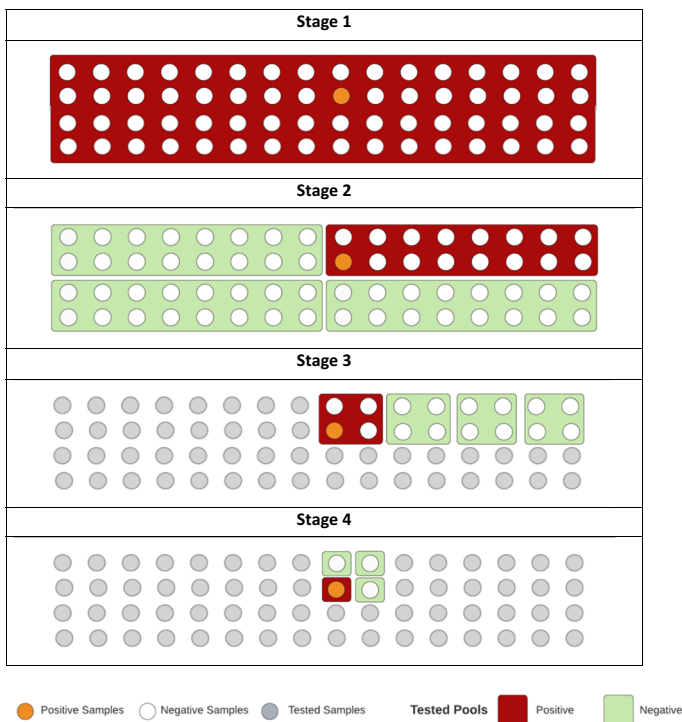


Fig. 3 Optimal grouping configuration for 64 samples in 4 stages

The results reported in Table 1 also indicate that the optimal number of kits for the multi-stage group partition problem follows nonincreasing order with stages. Because of this property, the proposed dynamic program can also be used to minimize the number of stages along with the minimization of the number of kits. Thus, the optimal solution to the group testing problem is the 4-stage group test, when the primary objective is to minimize the number of test kits, and the secondary objective is to minimize the required number of samples. The stage directly represents the maximum number of samples collected from an individual to identify the person infected with the coronavirus. Since every country is trying its best to flatten the curve of coronavirus cases, it is necessary to identify positive patients quickly. The number of stage minimizations is directly related to the time minimization to identify the positive case. The proposed MSGT model not only optimizes the total number of required kits but also reduces the time required to identify the positive report.

5.2 Numerical experiments for different sample sizes

This subsection provides the optimal solutions for different sample sizes capacity. The sample size capacity represents the ability of the country/province/organization/institute/laboratory to perform a group test with reasonable accuracy. Different reports available in public indicate that different organizations have developed methods to test the group pools of samples for a different maximum capacity. Therefore, we provide the optimal solutions for different sample sizes with the primary objective to minimize the number of test kits and secondary objectives to minimize the number of stages. We also report the solution for the existing Binary Tree group testing method for comparison purposes.

The results reported in Table 3 compare the solution in terms of the number of kits and the number of stages for the binary tree method and the multi-stage group testing method proposed in this paper. The results reported in Table 2 shows that our proposed method produces a better solution than the existing binary tree method for both the number of kits and the number of stages. For a sample size of 10, the MSGT method requires 8 kits and 3 stages while the binary tree method requires 9 kits and 5 stages. For a sample size of 99, the MSGT method requires 14 kits and 5 stages, while the binary tree method requires 15 kits and 8 stages.

The results reported in Table 3 can be used by an organization to determine the minimum total number of kits required and the optimal number of sample collections according to their ability to process group tests. If an organization can test the group of 10 samples, then they need 8 kits in 3 stages. They need to collect swab samples 3 times to identify the person infected with the coronavirus. If an organization has the capability to test the group of 99 samples, then they need just 14 kits and require 5 swab collections. The results reported in Table 3 also indicate that the required test kits increase with sample size. However, the increase in required kits with sample size is marginal. A sample size of 10 requires 8 test kits, while a sample size of 99 requires 14 test kits. The results indicate that an ability to test larger samples makes the coronavirus screening fast and efficient.

5.3 Managerial implications

The multi-stage group testing problem for coronavirus testing is an important and timely study to identify potential threats resulting from the COVID-19 pandemic. Our approach developed in this research can minimize the number of test kits required and the number of stages when identifying infected persons. The optimization is carried out using the dynamic

Table 3 The optimal solutions for different sample sizes

Sample size	Binary tree method		MSGT method		Sample size		Binary tree method		MSGT method	
	Best kits	Best stages	Best kits	Best stages	Best kits	Best stages	Best kits	Best stages	Best kits	Best stages
10	9	5	8	3	55	13	7	13	4	
11	9	5	8	3	56	13	7	13	4	
12	9	5	8	3	57	13	7	13	4	
13	9	5	9	3	58	13	7	13	4	
14	9	5	9	3	59	13	7	13	4	
15	9	5	9	3	60	13	7	13	4	
16	9	5	9	3	61	13	7	13	4	
17	11	6	9	4	62	13	7	13	4	
18	11	6	9	4	63	13	7	13	4	
19	11	6	10	3	64	13	7	13	4	
20	11	6	10	3	65	15	8	13	5	
21	11	6	10	4	66	15	8	13	5	
22	11	6	10	4	67	15	8	13	5	
23	11	6	10	4	68	15	8	13	5	
24	11	6	10	4	69	15	8	13	5	
25	11	6	10	4	70	15	8	13	5	
26	11	6	10	4	71	15	8	13	5	
27	11	6	10	4	72	15	8	13	5	
28	11	6	11	4	73	15	8	13	5	
29	11	6	11	4	74	15	8	13	5	

Table 3 (continued)

Sample size	Binary tree method		MSGT method		Sample size	Binary tree method		MSGT method	
	Best kits	Best stages	Best kits	Best stages		Best kits	Best stages	Best kits	Best stages
30	11	6	11	4	75	15	8	13	5
31	11	6	11	4	76	15	8	13	5
32	11	6	11	4	77	15	8	13	5
33	13	7	11	4	78	15	8	13	5
34	13	7	11	4	79	15	8	13	5
35	13	7	11	4	80	15	8	13	5
36	13	7	11	4	81	15	8	13	5
37	13	7	12	4	82	15	8	14	5
38	13	7	12	4	83	15	8	14	5
39	13	7	12	4	84	15	8	14	5
40	13	7	12	4	85	15	8	14	5
41	13	7	12	4	86	15	8	14	5
42	13	7	12	4	87	15	8	14	5
43	13	7	12	4	88	15	8	14	5
44	13	7	12	4	89	15	8	14	5
45	13	7	12	4	90	15	8	14	5
46	13	7	12	4	91	15	8	14	5
47	13	7	12	4	92	15	8	14	5
48	13	7	12	4	93	15	8	14	5
49	13	7	12	5	94	15	8	14	5

Table 3 (continued)

Sample size	Binary tree method		MSGT method		Sample size	Binary tree method		MSGT method	
	Best kits	Best stages	Best kits	Best stages		Best kits	Best stages	Best kits	Best stages
50	13	7	12	5	95	15	8	14	5
51	13	7	12	5	96	15	8	14	5
52	13	7	12	5	97	15	8	14	5
53	13	7	12	5	98	15	8	14	5
54	13	7	12	5	99	15	8	14	5

programming approach. The main challenge of implementing our method in real-life situation is the capability to test a group of samples with acceptable accuracy. The publicly available news article indicates that different countries have developed different methods to test group samples. For example, researchers from the Stanford School of Medicine have developed the capability to test 10 samples with reasonable accuracy, and they have successfully conducted group pool tests in the Bay Area of USA (Conger, 2020). Researchers at the German Red Cross Blood Donor Service in Frankfurt and the University Hospital Frankfurt have successfully tested 10 samples for coronavirus using PCR (polymerase chain reaction) procedure (Pool Testing of SARS-COV-02 (2020)). Researchers from the Israel Institute of Technology have successfully tested 64 samples using the PCR procedure (Cosh, 2020). The results obtained in this paper help to know the optimal group partition for different sample sizes, which depend on the country's capability to perform group tests.

Another managerial implication for our method is its ability to provide an optimal solution for a specified number of stages. The number of stages is directly proportional to the time taken to identify an infected person. Depending on the availability of resources and virus growth, the administration can specify the number of stages to identify an infected person. At the beginning of the pandemic (around mid-March 2020), many provinces in Canada took one day to get the results for the COVID-19 test. In this situation, 4 stages would have required 4 days to identify the infected person. During this period, positive cases were moving exponentially. In this situation, the administration could have specified the shorter stages (say 2 or 3 stages) for group testing. On the other hand, the administration can specify longer stages when the prevalence rate is low.

Even though the problem considered in this paper is limited to a single positive case within the group, it can easily be extended to assist in identifying multiple positive cases within the group. This approach can motivate new research in other multi-stage group testing problems to contain the spread of the coronavirus disease.

6 Conclusions and future research

In this paper, we have addressed the need of the hour and developed a dynamic programming method for multi-stage group testing for screening viruses during the COVID-19 pandemic. The proposed model advocates sampling different group sizes at different stages contrary to the existing binary tree method, which uses just two groups in every stage. It also minimizes the total number of test kits and the number of stages. The reduced number of test kits signifies the optimal use of resources, while the number of stages reduces the time required to identify the infected person. Both objectives ultimately help in the screening of coronavirus very quickly which is the most vital parameter for every country to save their economy caused by the pandemic lockdown. The approach developed in this paper also obtains an optimal solution for a specified number of stages. Our model facilitates a formulation that more closely conforms to real situations. This has been confirmed by several numerical examples that address the proposed approach and illustrate its applicability in real-life situations.

The method developed in this paper and the results obtained in this paper can be used for framing the policy of the group testing method because of its simplicity. The proposed policy for 64 samples group testing would be testing 4 groups with 16 members in the second stage, 4 groups with 4 members in the third stage and finally 4 individual testings. For other samples, similar policies can be framed using the results reported in our paper. If implemented

properly and timely, the proposed approach could prove to be a breakthrough in the proper utilization of testing resources and thus flatten the infection curve.

One limitation of our multi-stage group partition method is the assumption of a single positive case in the group. However, this paper provides a foundation for more general multi-stage group testing methods. Future research can analyze the case of more than one infected person in a group.

Funding The research is funded by: Canadian Network for Research and Innovation in Machining Technology, Natural Sciences and Engineering Research Council of Canada (Grant # 213090), and CN through the Centre for Supply Chain Management, Wilfrid Laurier University.

References

- Aldridge, M., Baldassini, L., & Johnson, O. (2014). Group testing algorithms: Bounds and simulations. *IEEE Transactions on Information Theory*, 60(6), 3671–3687.
- Aldridge, M. (2018). Individual testing is optimal for nonadaptive group testing in the linear regime. *IEEE Transactions on Information Theory*, 65(4), 2058–2061.
- Aloise, D., Hansen, P., & Liberti, L. (2012). An improved column generation algorithm for minimum sum-of-squares clustering. *Mathematical Programming*, 131(1–2), 195–220.
- Aprahamian, H., Bish, D. R., & Bish, E. K. (2019). Optimal risk-based group testing. *Management Science*, 65(9), 4365–4384.
- Assad, A., Wani, M. A., & Deep, K. (2020). A comprehensive strategy to lower number of COVID-19 tests. Available at SSRN 3578240.
- Babaki, B., Guns, T., & Nijssen, S. (2014). Constrained clustering using column generation. *International conference on AI and OR techniques in constraint programming for combinatorial optimization problems* (pp. 438–454). Springer.
- Ben-Ami, R., Klochendler, A., Seidel, M., Sido, T., Gurel-Gurevich, O., Yassour, M., Meshorer, E., Benedek, G., Fogel, I., Oiknine-Djian, E., & Gertler, A. (2020). Large-scale implementation of pooled RNA extraction and RT-PCR for SARS-CoV-2 detection. *Clinical Microbiology and Infection*, 26(9), 1248–1253.
- Biswas A. (2020). COVID-19: A way to test more people with fewer kits <https://science.thewire.in/the-sciences/covid-19-a-way-to-test-more-people-with-fewer-kits/>
- Braut, V., Mallein, B., & Rupprecht, J. F. (2021). Group testing as a strategy for COVID-19 epidemiological monitoring and community surveillance. *PLoS Computational Biology*, 17(3), e1008726.
- Brusco, M. J., Shireman, E., Steinley, D., Brudvig, S., & Cradit, J. D. (2017). Gaussian model-based partitioning using iterated local search. *British Journal of Mathematical and Statistical Psychology*, 70(1), 1–24.
- Conger K. (2020) Testing pooled samples for COVID-19 helps Stanford researches track early viral spread in Bay Area; <http://med.stanford.edu/news/all-news/2020/04/testing-pooled-samples-to-track-early-spread-of-virus.html>
- Cosh C. (2020): Israel researchers offer hope for more efficient coronavirus testing. <https://nationalpost.com/opinion/colby-cosh-israeli-researchers-offer-hope-for-more-efficient-coronavirus-testing>
- Deng, Q., Santos, B. F., & Curran, R. (2020). A practical dynamic programming based methodology for aircraft maintenance check scheduling optimization. *European Journal of Operational Research*, 281(2), 256–273.
- Delaigle, A., Huang, W., & Lei, S. (2020). Estimation of conditional prevalence from group testing data with missing covariates. *Journal of the American Statistical Association*, 115(529), 467–480.
- Dorfman, R. (1943). The detection of defective members of large populations. *The Annals of Mathematical Statistics*, 14(4), 436–440.
- Eberhardt, J. N., Breuckmann, N. P., & Eberhardt, C. S. (2020). Multi-Stage Group Testing Improves Efficiency of Large-Scale COVID-19 Screening. *Journal of Clinical Virology*, 128, 104382.
- Feng, J., Liu, L., & Parlar, M. (2010). An efficient dynamic optimization method for sequential identification of group-testable items. *IIE Transactions*, 43(2), 69–83.
- Gollier, C., & Gossner, O. (2020). Group testing against Covid-19. *Covid Economics*, 1(2), 32–42.
- Gossner & Gollier (2020). A temporary coronavirus testing fix: Use each kit on 50 people at a time, <https://www.washingtonpost.com/outlook/2020/03/31/coronavirus-testing-groups/>

- Hocking, T. D., Rigaiill, G., Fearnhead, P., & Bourque, G. (2020). Constrained dynamic programming and supervised penalty learning algorithms for peak detection in genomic data. *Journal of Machine Learning Research*, 21, 1–28.
- Ivanov, D. (2020). Viable supply chain model: integrating agility, resilience and sustainability perspectives—Lessons from and thinking beyond the COVID-19 pandemic. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-020-03640-6>
- Jessop, A. (2010). An optimising approach to alternative clustering schemes. *Central European Journal of Operations Research*, 18(3), 293–309.
- Jensen, R. E. (1969). A dynamic programming algorithm for cluster analysis. *Operations Research*, 17(6), 1034–1057.
- Kaniyamattam, K., Hertl, J., Lhermie, G., Tasch, U., Dyer, R., & Gröhn, Y. T. (2020). Cost benefit analysis of automatic lameness detection systems in dairy herds: A dynamic programming approach. *Preventive Veterinary Medicine*, 178, 104993.
- Khodare, A., Padhi, A., Gupta, E., Agarwal, R., Dubey, S., & Sarin, S. K. (2020). Optimal size of sample pooling for RNA pool testing: An avant-garde for scaling up severe acute respiratory syndrome coronavirus-2 testing. *Indian Journal of Medical Microbiology*, 38(1), 18.
- Li, T., Chan, C. L., Huang, W., Kaced, T., & Jaggi, S. (2014). Group testing with prior statistics. In *2014 IEEE International Symposium on Information Theory* (pp. 2346–2350). IEEE.
- Queiroz, M. M., Ivanov, D., Dolgui, A., & Wamba, S. F. (2020). Impacts of epidemic outbreaks on supply chains: mapping a research agenda amid the COVID-19 pandemic through a structured literature review. *Annals of Operations Research*. <https://doi.org/10.1007/s10479-020-03685-7>
- Rani, D., Mourato, S., & Moreira, M. (2020). A generalized dynamic programming modelling approach for integrated reservoir operation. *Water Resources Management*, 34, 1335–1351.
- Robbins, M. J., Jenkins, P. R., Bastian, N. D., & Lunday, B. J. (2020). Approximate dynamic programming for the aeromedical evacuation dispatching problem: Value function approximation utilizing multiple level aggregation. *Omega*, 91, 102020.
- Saltzman A. (2020). COVID-19 has world's major economies on track for worst quarterly decline in history, <https://www.cbc.ca/news/business/covid19-economy-jobs-oil-gdp-1.5533030>
- Saraniti, B. A. (2006). Optimal pooled testing. *Health Care Management Science*, 9(2), 143–149.
- Sinnott-Armstrong, N., Klein, D., & Hickey, B. (2020). Evaluation of group testing for SARS-CoV-2 RNA. *medRxiv* doi:<https://doi.org/10.1101/2020.03.27.20043968>
- Summers, D. S., Robbins, M. J., & Lunday, B. J. (2020). An approximate dynamic programming approach for comparing firing policies in a networked air defense environment. *Computers and Operations Research*, 117, 104890.
- Van Os, B. J., & Meulman, J. J. (2004). Improving dynamic programming strategies for partitioning. *Journal of Classification*, 21(2), 207–230.
- Voelkel, M. A., Sachs, A. L., & Thonemann, U. W. (2020). An aggregation-based approximate dynamic programming approach for the periodic review model with random yield. *European Journal of Operational Research*, 281(2), 286–298.
- Winston, W. L. (2004). *Operations research: applications and algorithms* (4th ed.). Thomson Brooks/Cole.
- WHO (2020). WHO Director-General's opening remarks at the media briefing on COVID-19—11 March 2020, <https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020>
- Xu, B., Rathod, D., Yebi, A., & Filipi, Z. (2020). Real-time realization of dynamic programming using machine learning methods for IC engine waste heat recovery system power optimization. *Applied Energy*, 262, 114514.
- Yelin, I., Aharon, N., Tamar, E. S., Argoetti, A., Messer, E., Berenbaum, D., Shafran, E., Kuzli, A., Gandali, N., Shkedi, O., & Hashimshony, T. (2020). Evaluation of COVID-19 RT-qPCR test in multi sample pools. *Clinical Infectious Diseases*, 71(16), 2073–2078.
- Zandkarimkhani, S., Mina, H., Biuki, M., & Govindan, K. (2020). A chance constrained fuzzy goal programming approach for perishable pharmaceutical supply chain network design. *Annals of Operations Research*, 295, 425–452.