npj | Science of Learning

www.nature.com/npjscilearn

ARTICLE OPEN

Check for updates

Optimized collusion prevention for online exams during social distancing

Mengzhou Li[1], Lei Luo[2], Sujoy Sikdar[3], Navid Ibtehaj Nizam[1], Shan Gao[1], Hongming Shan[1], Melanie Kruger[4], Uwe Kruger[1], Hisham Mohamed[1], Lirong Xia[2] and Ge Wang[1]✉

Online education is important in the COVID-19 pandemic, but online exam at individual homes invites students to cheat in various ways, especially collusion. While physical proctoring is impossible during social distancing, online proctoring is costly, compromises privacy, and can lead to prevailing collusion. Here we develop an optimization-based anti-collusion approach for distanced online testing (DOT) by minimizing the collusion gain, which can be coupled with other techniques for cheating prevention. With prior knowledge of student competences, our DOT technology optimizes sequences of questions and assigns them to students in synchronized time slots, reducing the collusion gain by 2–3 orders of magnitude relative to the conventional exam in which students receive their common questions simultaneously. Our DOT theory allows control of the collusion gain to a sufficiently low level. Our recent final exam in the DOT format has been successful, as evidenced by statistical tests and a post-exam survey.

npj Science of Learning (2021)6:5 ; https://doi.org/10.1038/s41539-020-00083-3

INTRODUCTION

Testing is essential for measuring and improving educational outcomes[1], but a major concern is that many students tend to cheat[2,3]. As suggested in a study between 2002 and 2015 by Dr. McCabe and the International Center for Academic Integrity[4], cheating among students was found astonishingly prevailing, e.g., 43%, 68%, and 95% of graduate students, undergraduate students, and high school students, respectively, admitted to cheating in assignments or exams.

Recently, the cheating problem has become much worse. In response to the COVID-19 pandemic, online learning has become necessary and exclusive in most educational systems[5,6]. The hard landing from the conventional education environment to the "emergency" online learning mode[7] creates various challenges, such as limited access to resources[8], lack of experience/skills[9,10], concerns over the quality and efficacy of education[6,11], as well as exacerbation of educational inequality[12]. As far as the assessment of learning outcomes is concerned, social distancing works directly against proctoring[13] since online testing performed at individual homes simply creates more chances to cheat[14] and increases temptation to do so[15–17]. Traditionally, physical invigilation is routinely used to suppress cheating. How to proctor online exams presents a new challenge during social distancing[6], as conventional approaches do not take the pandemic into account[14]. Rigorous online proctoring methods with cameras and associated technologies have been designed and used to prevent cheating[18] during the pandemic to effectively improve learning outcomes[19,20]. Professional services exist for online proctoring, such as TOP HAT[21] (used by over 400 institutions), Examity[22] and Proctortrack™[23] (proctored over two million exams). They monitor students through webcams and screen videos, enforce a full screen mode, and disable any content sharing. Some proctoring companies sign contracts with schools, while others charge students instead; as examples, ProctorU charges students $15 per test, while Proctorio charges a $100 lifetime fee. In addition to the costs associated with the use of third-party proctoring software, there are concerns over privacy[24–26]. What aggravates the problem of cheating is the "digital arms race", i.e., "finding new ways of cheating requires new ways to prevent it"[27].

Despite the benefit of rigorous proctoring, there is also a valid concern that using "such draconian measures" bluntly signals to our students the lack of our trust in their honesty[14]. Hence, in contrast to control the remote assessment environment, the OpenProctor system has been developed recently which extracts the writing style from learner-generated data and utilizes it as a behavioral biometrics to validate the authorship of students with machine learning[28]. This method demonstrated a mean accuracy of 93% significantly higher than the human performance baseline of 12%[29]. Unfortunately, the utility of this type of method is limited to text plagiarism and does not apply to multiple-choice and calculation questions, which are necessary and essential in majority science and engineering courses[15]. As mentioned in ref. [30], due to the highly objective nature of "math or fact-based" courses, it is more challenging and frequently questionable to maintain academic integrity without proctoring compared to the subjective "writing-based" courses. In addition, this writing-style recognition method mainly focuses on the post-exam stage, which may not be enough since it does not reduce the practicality of cheating and is not optimal as questioned by Fuller et al.[14]. ("Is Faculty's role to merely catch and punish cheating students or is it to support students through their studies so that ultimately, they can be confident that by working hard they will be successful without having to resort to deception?")

Besides such fancy techniques, traditional online learning experience also offers tips and recommendations without the use of cameras, which can be integrated to form a practical solution; e.g., sequencing questions randomly, presenting questions in limited time slots[31], and drawing assessment questions from a large pool[26,32].

[1]Department of Biomedical Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA. [2]Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, USA. [3]Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO, USA. [4]Department of Mechanical Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA. ✉email: Wangg6@rpi.edu

Published in partnership with The University of Queensland

npj nature partner journals

However, the transition from an emergency ad hoc remote assessment[7] to a valid conventional online assessment requires extensive efforts from educators; e.g., creating a large question pool. The pool size often needs to be huge to make the overlap of questions negligible between tests, e.g., a 300-question pool is needed for 30-question tests to control the average number of questions in common for two students below 3 (square of the number of questions in a test divided by the pool size)[15,26]. Such a pool is so large that is impractical to be updated frequently, which makes it vulnerable to cheating as evidenced by the rapid growth of exam questions being posted online during the pandemic.

Here we address the above limitations by providing an optimization-based cost-effective and privacy-conserving solution to help educators perform a valid remote assessment with minimal efforts. Specifically, the following three features of our approach are underlined: First, our method is optimization-based for simultaneously minimizing the productivity and practicality of cheating. The two models of the decision-making process behind cheating are well known: ref. [33] proposed a model involving the two competing processes of rational cost–benefit and effects on "self-concept", and ref. [34] developed a model based on the fraud triangle of incentive/pressure, rationalization, and opportunity. Our approach substantially increases the cost–benefit ratio and decreases the opportunity, and hence directly guides students to realize that it is more productive to finish the exam independently than to cheat. This curb on collusion is independent of proctoring, and respectful to privacy. Second, our method minimizes the question pool size; e.g., a pool of size 1.5 times the number of questions in a test is found to be sufficient to suppress collusion gain to an insignificant level with our method. The substantially smaller required pool size allows educators to devise their own questions relatively easily that require intellectual efforts than factual recalls that can be simply done via Google search[14], and update the questions frequently[31] rather than directly rely on published question banks without paraphrasing[32] which has a high risk of inviting academic dishonest[35–37]. Clearly, our framework encourages better compliance with best practices because of smaller question banks. Third, our method mainly focuses on thwarting collusion, which is believed to be significantly more popular than other types of cheating behaviors in online exams as found in a survey study based on self-reports[17] and validated later by direct measurements[38], showing that about 80% cheating events belonged to collusion, 42% showed copying from Internet website, and 21% fell into both categories. Other types of misconduct, such as accessing unauthorized sources and contract cheating, may also exist which can be addressed by incorporating readily available techniques; e.g., design open books questions[39–41], profile based authentication[42], challenging questions[43,44], and Web video conference proctoring.

In the following, we will focus on the key elements of our approach although the aforementioned complementary strategies are also important to complement our approach into an integrated practical solution to the anti-cheating problem. Our method is mainly designed for "math or fact-based" courses and compatible with most types of questions, and here is illustrated with a multiple-choice question (MCQ)-based model, since MCQs are popular, reliable, valid, and cost-effective[45,46]. Our main results are a theorem giving an upper bound of the collusion gain for our exam design, scheduling algorithms for anti-collusion in our distanced online testing (DOT) platform, and our DOT exam results. Using our DOT technology, the collusion gain can be practically and theoretically made insignificant, especially by incorporating prior knowledge of the students' competences. The collusion gain refers to the percentage score increased by a student through collusion, and competence represents the student's individual probability by which he/she can correctly answer questions in an exam. Our main idea is to optimally deliver questions to students as individual-specific sequences in a synchronized fashion so that even if students freely cheat among themselves they still cannot significantly improve their scores (Fig. 1).

RESULTS

Theorem bounding the collusion gain

As a first order of approximation, our analysis is focused on an idealized DOT scenario, but our analysis can be extended to more general settings without theoretical or technical difficulties. In our initial DOT setting, M_1 MCQs from a pool of M_2 MCQs (for example, with equal difficulty and credits for convenience, which can be readily relaxed for a more accurate analysis) are provided to a class of N students, and there are Q choices per question with one being correct. All N students are presented with their own set of M_1 questions displayed one by one in generally different *sequences*, and are asked to take the exam simultaneously. Each student must answer each question in a predetermined time slot, and cannot revisit previous questions. This mode of delivering questions is exemplified in Fig. 1a.

Under practical assumptions on students' collusion behaviors ("Methods"), we propose a grouping-based anti-collusion scheme (GAS) to control the collusion gain below any desired level with prior knowledge on students' competences. The competence of a student can be easily estimated based on his/her grade point average (GPA) (rough surrogates), from earlier quizzes (better indicators), and/or with a first portion of the exam (achievable via dynamic programming). Generally speaking, our grouping-based approach consists of the following three elements: (1) *Grouping*: Students with similar competences are grouped together to receive the same sequence of questions in an exam; (2) *Optimization*: The number of questions that can be copied between groups is aggressively reduced (even to zero coupled with the next element); (3) *Augmentation*: The pool of questions can be enlarged to have the number of questions greater than M_1.

The anti-collusion exam design can efficiently reduce the collusion gain mainly due to following reasons (Fig. 1b–d): (1) The maximum question leakage from top to down of C consecutive cyclic sequences can be reduced to zero if $M_2 - M_1 + 1 \geq C$ (Supplementary Fig. 1); (2) by grouping, the equivalent number of students (the number of groups) can be significantly reduced to just use the C sequences; (3) students with similar competences have small probabilities to cheat within their group due to the fact that they can only obtain tiny collusion gains, although the intra-group collusion is facilitated because of the same sequence shared. With this procedure, by making $C = M_2 - M_1 + 1$ sufficiently large we can control the maximum individual collusion gain as well as the average collusion gain below any desired level.

Mathematically, we present the following theorem that shows the upper bound of the collusion gain associated with our GAS (Supplementary Note 1).

Theorem 1. *Given sequences of M_1 questions from the bank of M_2 MCQs with one and only one correct choice out of Q choices for each question, the maximum individual collusion gain can be controlled to be no larger than $(1 - 1/Q)/(M_2 - M_1 + 1)$ using the GAS.*

This theorem is practically powerful; e.g., according to this upper bound, the maximum individual collusion gain can be theoretically controlled below 3.6% for any large-size class with a reasonable test setting of $M_2 = 60, M_1 = 40, Q = 4$.

Metrics characterizing the final exam design

Our aforementioned theorem provides an upper bound for collusion control, but it is usually not optimal since it does not fully take advantage of the knowledge of students' competences.

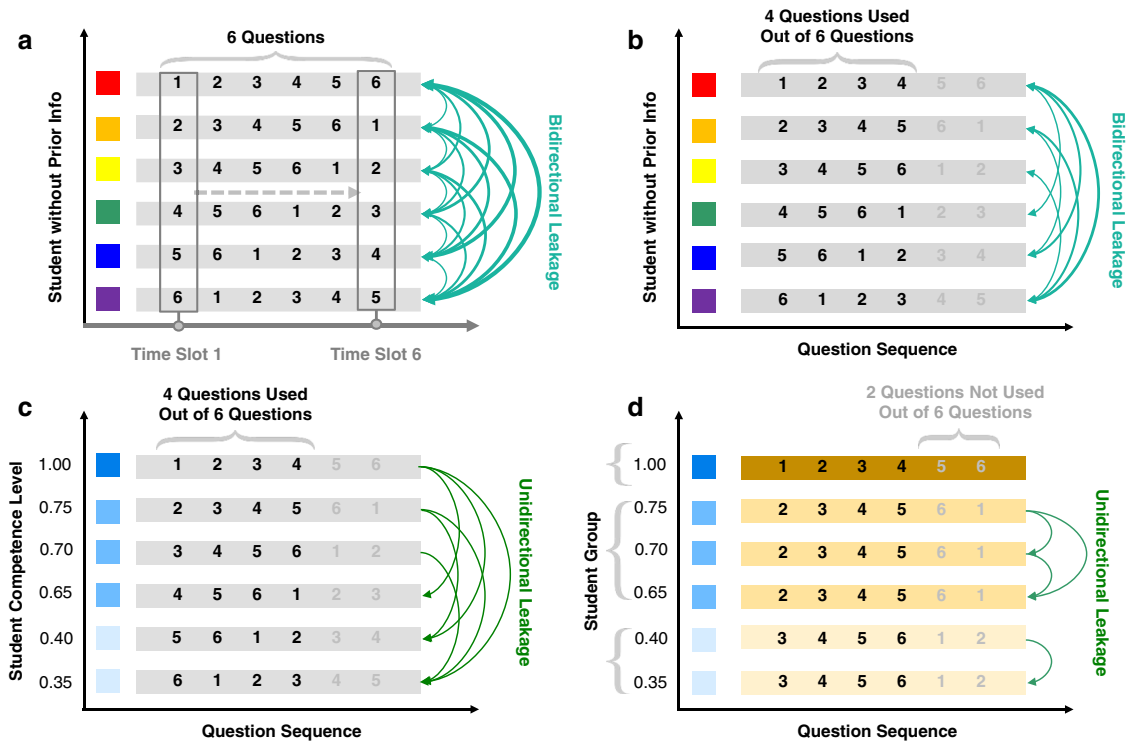

Fig. 1 Anti-collusion mechanisms with and without prior knowledge of students' competencies. Assume that collusion happens between two students and one can get answers from the other on questions that the other has already answered or is working on (see "Methods"). **a** The circulation-based scheme is illustrated with a simple example, in which six students take an exam consisting of six questions ($M_1 = M_2 = 6$) provided to each student one by one, and each question must be finished within the allocated time slot shown as the vertical box. If there is no information of students' competences, this scheme helps reduce potential bidirectional cheating among students to ~50% of question; **b** the collusion chance can be made even less if cheating students are fed with more new questions ($M_1 = 4, M_2 = 6$); **c** if prior information on students' competences is available, the naive assignment in **b** still yields significant collusion gains; but **d**, using our grouping-based anti-collusion scheme, the maximum and average collusion gains can be sharply reduced to ~10% and ~3%, respectively. The scheme first divides the competence range into $M_2 - M_1 + 1$ intervals, then groups the students into these intervals properly, finally assigns these groups of students with the corresponding number of consecutive cyclic sequences, respectively. The maximum collusion gain with this scheme is bounded by our Theorem 1.

Based on the results of GAS, discrete optimization algorithms ("Methods") can be used to further reduce the collusion gain for the best DOT anti-collusion performance. For this purpose, the objective function needs to be defined as follows.

Let us introduce the *competence profile* of students $Y = \{y_i \in [1/Q, 1] |\ i = 1, 2, \ldots, N\}$ in a non-increasing order, and a *colluding matrix* $P = (p_{j,i})_{i,j\in[N]}$, where $p_{j,i}$ represents the probability of student i colluding from student j if $i \neq j$, and $p_{i,i}$ the probability that student i does not cheat in the exam. P is upper triangular. Given an *assignment* $A = (a_i, \ldots, a_N)$ which is a vector whose elements are sequences of questions (SQs), where a_i is the SQ assigned to student i, the average collusion gain g is the total collusion gain normalized with respect to the class size and the number of questions in an exam, and defined as

$$g(A) = \frac{\text{sum}\{Z(A) \circ P \circ D\}}{NM_1} = \sum_{i=1}^{N}\sum_{j=1}^{i-1}\frac{z_{j,i}(A)}{NM_1}p_{j,i}(y_j - y_i) \quad (1)$$

where sum{·} stands for the operation of summing up all elements, ∘ denotes the Hadamard (element-wise) multiplication, the *competence difference matrix* D is defined as $(d_{j,i})_{i,j\in[N]}$ where $d_{j,i} = \max(y_j - y_i, 0)$, and the *positional matrix*$Z = (z_{j,i})_{i,j\in[N]}$ is determined by A where $z_{j,i}$ represents the number of questions that student i can cheat from student j if $j \neq i$, and the special case $z_{i,i}$ is defined as M_1. If all students use the same SQ as in the conventional exam scenario without collusion prevention, the average collusion gain becomes

$$g_0 = \frac{\text{sum}\{P \circ D\}}{N} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{i-1}p_{j,i}(y_j - y_i). \quad (2)$$

We developed our DOT platform (Supplementary Note 6) incorporating the anti-collusion techniques as well as other complementary techniques for online exams, and applied this platform for the final exam of an undergraduate imaging course on 28 April 2020. Totally, 78 out of 85 undergraduate students took the exam from two separately taught classes. The exam consisted of $M_1 = 40$ questions that were assigned to each student and scheduled from a pool of $M_2 = 60$ questions by applying our greedy algorithms with a heuristically constructed colluding matrix P, detailed in the "Methods" section and Supplementary Note 4. During the exam, the students were asked to join a WebEx session for the instructors to address any questions or technical difficulties (in principle, our DOT technology can be combined with sounds online proctoring for an enhanced performance at an additional cost). The competency information of the students was estimated based on their performance in the midterm exam conducted before the class was taught online.

The optimized assignments led to orders of magnitude reduction in the collusion gain. Quantitatively, the average collusion gain was reduced to 0.0073% from 19.23% (a reduction by three orders of magnitude from the conventional scenario), with the worst-case collusion gain (g_W, the average collusion gain when every student manages to achieve his/her maximum possible collusion gain; see "Methods") and the maximum individual collusion gain (g_{MI}, the

Table 1. Collusion gain estimation and optimization in the case of $N = 85$, $M_2 = 60$, $M_1 = 40$, and $Q = 4$.

Condition	Anti-collusion	Mean (standard seviation)		
		g	g_W	g_{MI}
(a) Collusion gain estimation of the optimized assignment (500 instances)[a]				
Accurate Y, random P	None	0.14497 (0.00139)	0.30901 (–)	0.75000 (–)
	Optimized	**0.00044** (0.00005)	**0.00908** (–)	**0.06878** (–)
Noisy Y, random P	None	0.14884 (0.00450)	0.30837 (0.00708)	0.73524 (0.02107)
	Optimized	**0.00190** (0.00052)	**0.06275** (0.00775)	**0.20404** (0.02941)
(b) Optimized performance over 500 random Y profiles[b]				
Random Y, heuristic P	None	0.16278 (0.01499)	0.30384 (0.04153)	0.60838 (0.06344)
	Optimized	**0.00007** (0.00002)	**0.00903** (0.00158)	**0.04970** (0.01650)
(c) Optimizations of different class sizes over 500 random Y profiles with heuristic P[c]				
$N = 20$, $M_2 = 30$, $M_1 = 20$	Optimized	0.00013 (0.00007)	0.00657 (0.00223)	0.03704 (0.01812)
$N = 40$, $M_2 = 60$, $M_1 = 40$	Optimized	0.00003 (0.00002)	0.00433 (0.00130)	0.02686 (0.01356)
$N = 100$, $M_2 = 60$, $M_1 = 40$	Optimized	0.00008 (0.00002)	0.00936 (0.00139)	0.04847 (0.01777)
$N = 500$, $M_2 = 60$, $M_1 = 40$	Optimized	0.00011 (0.00001)	0.01400 (0.00086)	0.07886 (0.01629)

[a]Robustness of the optimized assignments: The collusion gain of assignments optimized with the heuristic P in which the colluding probability is proportional to the competence difference between two students (see "Methods") is reproduced in two kinds of perturbations: noisy Y (Gaussian noise ($\mu = 0$, $\sigma = 0.05$) on accurate Y) and P variations (random colluding probabilities following the Dirichlet distribution). [b]Stability of the optimized performance: The optimization results over 500 random Y profiles, each of which was randomly generated according to a Gaussian distribution ($\mu_0 = (1 + 1/Q)/2$ and $\sigma_0 = (1 - 1/Q)/6$ on the support $[1/Q, 1]$). [c]Optimization performances on small-size classes ($N = 20$, $M_2 = 30$, $M_1 = 20$, and $Q = 4$), middle-small-size classes ($N = 40$, $M_2 = 60$, $M_1 = 40$, and $Q = 4$), middle-size classes ($N = 100$, $M_2 = 60$, $M_1 = 40$, and $Q = 4$), and large-size classes ($N = 500$, $M_2 = 60$, $M_1 = 40$, and $Q = 4$). Bold indicates the better result.

maximum of the maximum possible collusion gains over all students; see "Methods") being 0.91% and 6.88%, respectively. Specifically, we performed numerical simulation to estimate the average collusion gain with optimized assignments under the following conditions: (1) accurate Y and random P, the estimated Y was assumed to be faithful and the colluding probabilities $p_{k,i}$ ($k < i$, $i - 1$ in total) assumed to follow the $(i - 1)$-variate *Dirichlet* distribution with a concentration parameter of $\alpha = 10$; (2) noisy Y and random P, the estimated Y was assumed to contain a Gaussian noise ($\mu = 0, \sigma = 0.05$). We calculated the average collusion gains g as well as the worst-case metrics g_W and g_{MI} with the same assignments (the conventional scenario) and with our optimized assignments over 500 instances for each condition. The resultant means and standard deviations demonstrate the accuracy and robustness of our DOT technology (Table 1a).

To further illustrate the performance of our DOT technology, in the setting of the above practical case we performed numerical simulations assuming random Gaussian-distributed competence profiles with $\mu_0 = (1 + 1/Q)/2$ and $\sigma_0 = (1 - 1/Q)/6$, truncated to be meaningful $[1/Q, 1]$ and heuristically constructed P from Y. We calculated the collusion gains without collusion prevention g_0 and with optimized prevention g over 500 instances for each configuration. Our results are summarized in Table 1b. It can be observed in this case that the mean of the average collusion gain can be reduced by three orders of magnitude with tiny standard deviations, suggesting that our DOT designs are not only effective but also stable in controlling the collusion gain. We further changed the number of students to those of four typical class sizes from $N = 20$ to $N = 500$ as shown in Table 1c, and the mean of the average collusion gain remains at a very small level which implies the practical applicability of our method in dealing with a wide range of class sizes.

Analyses on the final exam

We first look at the final exam results, which are summarized in several histograms. The normalized distribution (zero mean, unit std.) of the 78 students' scores out of 40 questions is, as expected, an approximate "bell-shaped curve" of a normal distribution (Fig. 2a). As a first comparison by eye, we contrast the distribution of the final exam results with that of the midterm exam, which serves as a control group here. For a more quantitative analysis, we applied standard tests to the results of the midterm and final exams to ascertain whether there are any anomalies embedded within the two sets. First, we found that both sample sets were drawn from normal distributions by applying the Anderson–Darling test[47] ($p = 0.1570$ and $p = 0.3004$ for the midterm and final samples, respectively). Next, we confirmed that both sample sets were drawn from the same normal distribution using the two-sample Kolmogorov–Smirnov test[48] ($p = 0.1574$). As an additional test, we applied the two-sample t-test for equal variance[49] and confirmed that the two distributions have the same mean ($p = 0.7997$). In summary, the evidence does not support the claim that there are differences in the distributions of the midterm and the final exam, demonstrating consistent evaluative results of the same population between the conventional physical proctoring method (the midterm) and our DOT format (the final).

Quantitatively, the maximum gain of the students through collusion is theoretically controlled by design to be below 7% (Fig. 2b). This compares favorably to a maximum gain of 75% without the use of our optimized anti-collusion technique. It is important to note that over 90% of students may have a maximum collusion gain of below 2%, which underpins the effectiveness of our technique. One feature of this technique is that not all student shared the same question sets, which helped to reduce the colluding chances between students. In terms of the number of recipients of each MCQ, only 19 questions were assigned to all students, and 20 questions were assigned to fewer than 40 students each (Fig. 2c).

Following from the preceding discussion, utilizing our anti-collusion exam design the controlled collusion gain was made very small but is still not zero. It is therefore imperative to test whether significant collusion did occur. To do so, we examined the following two aspects: (i) what is the frequency with which pairs of students gave the same incorrect answer and (ii) is the average number of correct answers to the first 20 questions comparable to

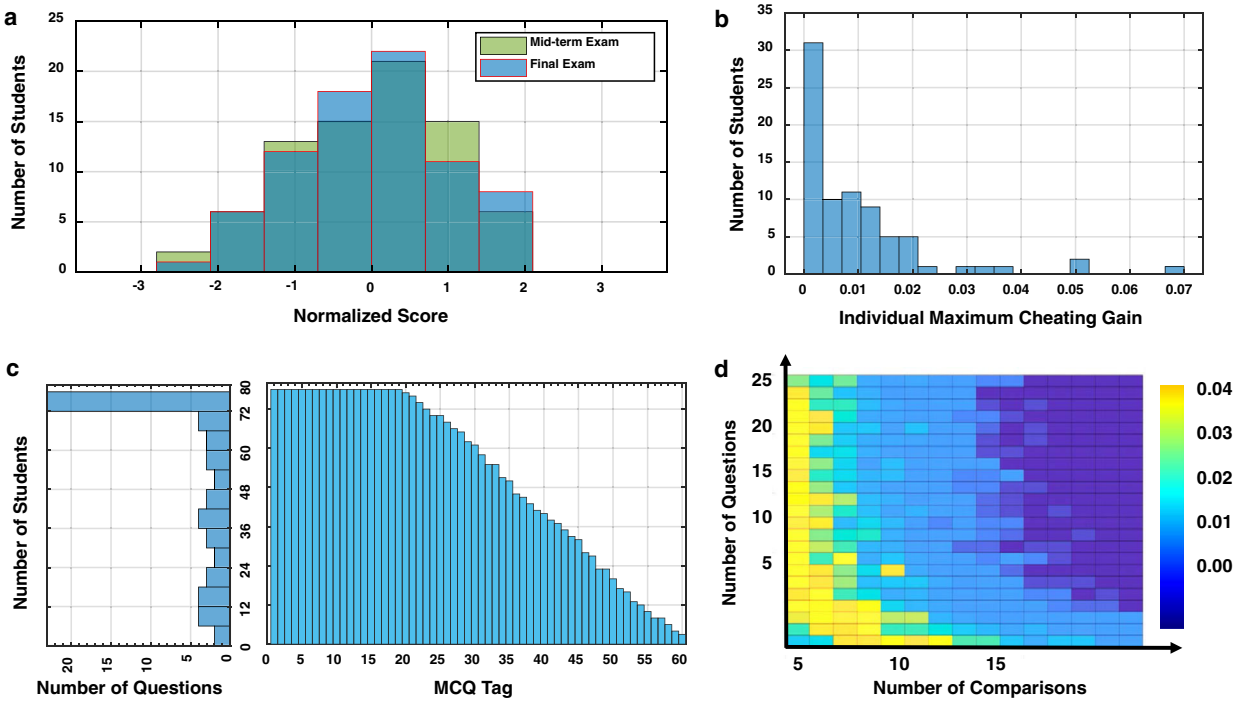

Fig. 2 Results of the final exam. a The distribution of the final exam scores (78 students, conducted on the DOT platform) in a normal 'bell' shape with insignificant differences from the distribution of the midterm exam scores (before social distancing), which indicates the effectiveness of our DOT technology in collusion prevention. **b** With our optimized question assignment, the distribution of calculated maximum possible collusion gains in terms of percentage score showing the maximum collusion gains strictly below 7%. **c** The histogram of the size of a particular set of MCQs versus the number of students who received this set of questions on the left, and the distribution of this number versus the MCQ tag on the right, showing that not all student received the same questions. **d** Plotting q values against the number of comparisons and the number of questions to test the hypothesis that collusion resulted in students giving a significantly higher number of the same incorrect answers. As all q values are below the significance level of 0.05, we conclude that no significant collusion occurred.

that of the last 20 questions. The rationale for aspect (i) is that the events of student pairs giving the same incorrect answers are random and independent if no collusion occurred. The basic premise of our test is the probability that two students gave the same answer for an MCQ with $Q = 4$ choices is 1/4, assuming that students' answers are independent. Conversely, this probability would be significantly higher if significant collusion occurred. The rationale for aspect (ii) is that the difference between the probabilities of correctly giving answers to the first and last 20 questions should also be random and on average zero if there was no collusion. On the other hand, given that collusion is more likely to occur during the latter half of the exam, we would expect an increase in the number of correct answers for the last 20 questions.

For aspect (i), we formulated and tested the hypothesis that significant collusion occurred using a set of paired tests[50]. The results of the hypothesis for testing aspect (i) confirm that the corresponding values for each false discovery rate is below t he significance of 0.05 (Fig. 2d). Therefore, the empirical evidence does not support that there was an abnormal number of student pairs who consistently gave identical incorrect answers (Supplementary Note 5). To address aspect (ii), we formulated and tested the hypothesis that the difference in means of correctly giving answers to the first and the last 20 questions is zero. Based on the 78 students' answers to their questions in the exam, we utilized the non-parametric Wilcoxon signed-rank test for paired observations[49], which yielded a p value of 0.3133. Based on the evidence, we cannot reject the hypothesis that the average numbers of correct answers to the first and the last 20 questions are identical. In other words, the difference is not statistically significant between the average numbers of correct answers to the first 20 questions and the last 20 ones.

Feedback from the post-exam survey

The post-exam survey indicates that the online exam using the DOT platform was well received by a majority of students (Fig. 3). More precisely, 76.9% of students (Fig. 3a) rated the duration for answering questions to be 3 or above out of a 5 point scale ranging from Very Insufficient (1) to Very Sufficient (5), and 80.8% students (Fig. 3b) rated the convenience of using the platform's interface to be 3 or above on a 5 point scale ranging from Very Inconvenient (1) to Very Convenient (5). The survey also secured feedback concerning the degree of difficulty for the exam questions. Close to 70% of students voted "reasonable", which is the third choice (Fig. 3c). When excluding the extremes "easy", or choice number 1, and "difficult", or choice number 5, 96.1% of students found the questions within the acceptable range (between 2 and 4). The survey finally inquired how similar the final online exam for the students was compared to other online exams they took. The students' opinions on how familiar the other online exams were with the look-and-feel of our online exam, showing that around 59% of students answered 3 or above out of a 5 point scale ranging from Very Different (1) to No Different (5) (Fig. 3d). The remaining 41% of students indicated that the format of the final exam is different to other exam settings by selecting options (1) and (2).

DISCUSSIONS

Although our method is only illustrated with MCQs, our method is actually compatible with most types of questions (except the easy-writing-type tests) since it is the optimized SQs that inhibit the collusion gain. Not to mention that many other types of questions can be easily adapted to the MCQ form. It is also worth noting that our method is compatible with other advanced techniques such

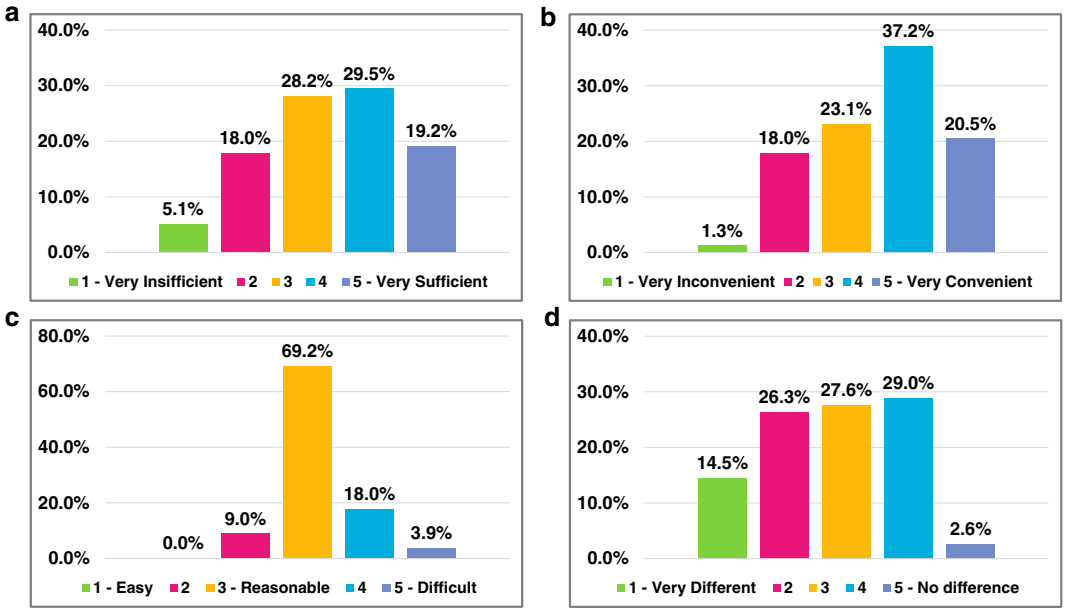

Fig. 3 Post-exam online survey results. Bar graph summary in terms of **a** sufficiency of time slot length, **b** convenience of platform interface, **c** easiness of exam questions, and **d** similarity to other online exams respectively (performed on 28 April 2020 for the undergraduate medical imaging course offered at Rensselaer Polytechnic Institute, Troy, New York, USA).

as "learning analytics"[29], which can be integrated into our method for text plagiarism detection in writing-based constructed-response questions.

In the aforementioned post-exam survey, we have received constructive comments from students on how to improve our DOT-format exam design. Specifically, we plan to make the DOT platform more flexible so that questions can have different weights/credits/time-lengths, and both the number of choices and the number of correct choices can be adjusted. Another potential adjustment is to use a soft moving window approach instead of presenting students with one question at a time. Within the soft moving window, a student can work on a small number of questions and amend the answers as needed within the moving time window. It is important to note that such extensions can be similarly analyzed in the discrete optimization framework and do not present any technical difficulty. When prior knowledge of students' competences is unavailable, an initial phase of an online exam can be devoted to estimate the students' competence levels. This is then followed by scheduling SQs for the remainder of the exam based on the estimated competence. Finally, the block chain technology[51] is highly relevant for keeping the database of questions confidential (accessible to faculty only) and managing students' individual educational credits systematically. These and other improvements can be readily implemented in our optimization framework.

In a recent study[52], it was shown that COVID-19 could be persistent for years, since thousands of mutations have happened (for example, a SARS-CoV-2 protein had 56% of its genes mutated), explaining many false-negative tests. Over the past several days, the United States experienced a reemerging first or second wave of newly diagnosed cases producing a significantly larger number of infections. Hence, social distancing and equivalent policies due to COVID-19 may remain in place in the near future or even over a longer period of time[53]. A positive response to the pandemic is to let online learning and testing practice enter the mainstream of educational activities or at least it can be assumed to play a significant role while it is being continuously improved. Thanks to the internet and computing technologies, high-quality DOT systems are now feasible solutions in offering comparable exam outcomes that are cost-effective and do not invade students' privacy.

In conclusion, we have proposed a new type of anti-collusion approach for online exams, which relies on discrete optimization in the permutation space and prior knowledge on students' competences to suppress collusion behaviors among students. Together with other complementary methods, the general cheating prevention purpose can be achieved. Also, we have reported our DOT platform and its successful application. It has been theoretically, numerically, and experimentally demonstrated that using the DOT technology allows reducing the cheating benefit cost-effectively so that accurate and reliable exams are feasible during social distancing and beyond.

METHODS

Assumptions on collusion behaviors

The assumptions on collusion behaviors are as follows:

1. Cheating is unidirectional. If two students A and B collaborate on collusion, and A has better competence than B, then only B will copy answers from A which is termed as B cheating from A and A helping B.
2. B can get the answer from A if A has already answered the question before B or they are working on the problem at the same time. Thus, different relative SQs (we denote 'sequence of questions' as SQ for short) for A and B will influence the number of questions that B can copy from A.
3. Each student can only cheat from no more than one student ("A helping B" model); Given the limited duration of the time slot for each question and stress involved during an exam, B is expected to rely on typically only one helper A. Put differently, as B requires assistance, he/she is not good at judging which answer is correct when different inputs come from multiple helpers (unless B uses a voting strategy which may or may not make a significant difference to his/her final score). Hence, we consider the "A helping B" model as reasonable in this context.
4. B can help C while cheating from A.
5. B cheating from A does not influence D cheating from A; in other words, one student can help multiple students.
6. An answer based on cheating is not disseminated further to help other students. This assumption can be justified by the argument that given the limited time of an exam and involved stresses, B is unlikely to remember what he/she copied from A and to have the time to provide C with the answer.

Estimation of students' competences

The students' competences are estimated based on their performance in the midterm exam before implementing social distancing. The two classes were taught by different instructors, and have different midterm exams, but they will take the same final at the same time. Thus, their relative performances in the class were treated as their competence score rather than their real scores. The grade distributions of the two classes were first normalized to the distribution with zero mean and unit variance, and then combined together. It is worth mentioning that students who did not participated in the midterm exam were excluded from the normalization procedure, and then put back to the combined profile with 0 (assigning an average performance to estimate their performance). Finally, combined normalized grades were then linearly transformed to the range [0.25, 1] to form the prior knowledge of the competence profile Y of the combined set of the students. Note that the range [0.25, 1] is empirically selected. Based on our experience, all questions were covered in the class and a few students got nearly perfect scores while a few totally unprepared students were also seen every semester. In addition, the heuristic colluding matrix P relies on the competence differences rather than the competence values, hence, the linear transformation of the competence range will only impose a constant scaling factor on the average collusion gain g base on equation (1), which will not influence the optimization result of the SQ assignments.

Construction of the colluding matrix

To perform the optimization, we heuristically construct a colluding matrix P depicting the probability of every student cheating from another student. Following the notation in the main text, reasonable assumptions about colluding mechanisms are made as follows: (1) The probability of student i actively cheating is related to his/her competence y_i; Student 1 tends not to cheat since he/she could obtain no gain (risk greater than benefit), while student N will try all means to cheat since he/she will always gain (benefit greater than risk). (2) The probability of collusion happens between two students A and B is related to the difference of y_A and y_B. Student i will have the strongest willingness to cheat from student 1, but the least willingness to cheat from student j if $y_i = y_j$ since he/she cannot trust j more than himself/herself, and he/she will never cheat from j if $y_i > y_j$.

Based on the assumptions above, the colluding matrix P is heuristically constructed as follows:

$$p_{j,i} = \begin{cases} 0, & y_j \le y_i \\ \frac{y_j - y_i}{\sum_{k=1}^{n_f(i)} (y_k - y_i)} (1 - p_{i,i}), & y_j > y_i \end{cases} \tag{3}$$

$$p_{i,i} = \left[1 - \frac{\sum_{k=1}^{n_f(i)} (y_k - y_i)}{\sum_{k=1}^{N} (y_k - y_N)} \right]^{\eta} \tag{4}$$

where $n_f(i)$ is defined as the number of elements in Y that are greater than y_i, and η is a non-negative constant which can be used to adjust students' willingness to cheat. Larger η will increase the colluding probability, and students are supposed to always commit active cheating if $\eta = \infty$ (all optimizations were conducted with this setting). Equations (3) and (4) define the probabilities of the cheating and non-cheating states of student i respectively, and in the cheating state, the possibility of student i will cheat from student j is proportional to their competence difference $y_j - y_i$ normalized by the sum of competence differences in all possible cases.

Without loss of generality, we further assume that students have different competences ($y_1 > y_2 > \cdots > y_N$), due to the fact that adding tiny differences to two equal y has a negligible effect on the result of g and simplify the expression of $n_f(i)$ to be of the form

$$n_f(i) = i - 1 \tag{5}$$

Hence, $p_{j,i}$ can be written more explicitly as follows:

$$p_{j,i} = \begin{cases} 0, & j<i \\ (1 - p_{i,i})(y_j - y_i)/(\sum_{k=1}^{i} y_k - i y_i), & j>i \\ \left[1 - \sum_{k=1}^{i} (y_k - y_i) / \sum_{k=1}^{N} (y_k - y_N) \right]^{\eta}, & j = i \end{cases} \tag{6}$$

Note that the heuristic colluding matrix P represents a practically reasonable start for optimization. We construct P to place a larger weight on the collusion between students with a larger competence difference than that with a small competence difference, which helps limit the collusion gain in the worst-case scenario. Since mismatches exist very likely between the model and the practice, any optimization result needs to be subjected to a worst-case analysis.

Analysis of worst-case metrics

Similar to the average case analysis and worst-case analysis in computer science, we may want to revisit our optimized results in a worst-case study since mismatch is very likely to exist between the model and the practice. Hence, another two important metrics are introduced to assess the optimization results from the risk control angle, i.e., the worst-case average collusion gain g_W defined as the average collusion gain in the situation where all students manage to achieve their maximum possible collusion gain (the maximum possible collusion gain of the student i is achieved by setting the probability of i cheats with the student j to 1, from whom i will obtain the maximum gain among other choices of i),

$$g_W(A) = \frac{1}{NM_1} \text{sum} \{ \max_{j \in [N]} \{ Z(A) \circ D \} \} \tag{7}$$

and the maximum individual collusion gain g_{MI} which is the maximum of the maximum possible collusion gains over all students,

$$g_{MI}(A) = \frac{1}{M_1} \max_{i,j \in [N]} \{ Z(A) \circ D \}. \tag{8}$$

g_W can be used to assess the performance of the optimized results under the worst situation and can be treated as a reliable upper limit estimation of the collusion gain under the given competence profile Y since the calculation of g_W does not involve the colluding matrix. g_{MI} is a metric can be used to estimate the fairness of the exam from the aspect of the maximum collusion gain any student can achieve. If the collusion gain calculated in the worst situation for the output assignment is not acceptable, the result should be used with caution or just change the initialization and generate more solutions. Overall notations and metrics of the model are summarized in Supplementary Tables 1 and 2.

Cyclic greedy searching

In principle the optimal assignment to achieve the minimized collusion gain should be searched from the set of all possible assignments whose size is n^N and n is the size of the pool of SQs P_{SQ}. Practically, an optimal solution will be computationally infeasible (seemingly NP-hard) if there are many students and/or many questions in the exam, hence we propose the following efficient algorithm. We first narrow the searching pool of SQs to the sequences generated by circular shifting (let us denote the set as P_{CS}) from P_{SQ} following the heuristic that P_{CS} is a good representative subspace of P_{SQ}. P_{CS} contains all possible z values achieved by any two sequences from P_{SQ}, and if we randomly choose two sequences from the two space, the expected z value of two sequences from P_{CS} is even smaller than that from P_{SQ} (see Supplementary Note 7 for the proof). Then, we choose to use a greedy-searching algorithm from a randomly initialized assignment or the assignment generated with the result of GAS, and repeat the searching process for multiple times until the loss does not decrease. Through this greedy searching, satisfactory results can be easily obtained in polynomial time.

Specifically, we can perform Greedy Searching from a Cyclic pool (Cyclic Greedy Searching, CGS). The concept behind CGS is to iterate with respect to each and every student, and replace his/her current sequence of questions with one from P_{CS} if the updated assignment achieves a smaller average collusion gain. Several cycles of greedy-searching are needed to fulfill a complete search, and the output assignment from the last cycle will be treated as the initialization for the next cycle during the iteration. We use the result from GAS as our preferred initialization, and other initialization is also suggested to be adopted and find the best one among the results to improve the solution (see Algorithm 2 in Supplementary Note 2 for pseudocodes and implementation details).

Min–max greedy matching

Instead of searching in the cyclic pool P_{CS}, we can search from the entire permutation pool P_{SQ} to minimize the collusion gain. Due to hardness of searching in a huge permutation space when the problem scale is large, we adapted the min–max greedy matching algorithm (MMM) to work in polynomial time. A natural approach is to start with an initial random assignment and improve it greedily by picking up one student at a time according to a certain order, and refining his/her SQ so that the total gain is minimized from the set of all possible M_1-permutations of M_2. We propose MMM to greedily improve an assignment, and show that computing a sequence to replace a single student's sequence in an assignment that minimizes the total gain can be done in polynomial time by performing a minimum weight maximum matching (see Algorithm 3 in

Supplementary Note 7 for implementation details). For convenience, we first introduce some notations. Given any $s \in P_{SQ}$:

1. For each $j \in [M_2]$, we define $s(j) = l$ if j appears in the lth position in s, and $s(j) = 0$ otherwise.
2. For each $j \in [M_2]$, $\alpha(s, j) = 1$ if $s(j) \geq 1$, and $\alpha(s, j) = 0$ otherwise, to indicate whether question j is on sequence s.
3. For each $j \in [M_2]$, each $l \leq M_1$, $\beta(s, j, l) = 1$ if $s(j) \geq 1$, $s(j) \leq l$, and $\beta(s, j, l) = 0$ otherwise, to indicate whether question j appears at or before position l on sequence s.
4. For each $j \in [M_2]$, each $l \leq M_1$, $\gamma(s, j, l) = 1$ if $s(j) \geq l$, and $\gamma(s, j, l) = 0$ otherwise, to indicate whether question j appears at or after position l on sequence s.
5. For any $s, s' \in P_{SQ}$, and any $j \in [M_2]$, $\delta(s, s', j) = 1$ if $s(j) > 1$, $s'(j) > 1$, and $s'(j) \leq s(j)$, and $\delta(s, s', j) = 0$ otherwise to indicate whether a student assigned s can cheat on question j from a student assigned s'.

Given an instance $([N], [M_2], [M_1], Y)$, MMM is initialized with an assignment A, and proceeds to greedily improve A in N rounds, one student at a time, as follows: In each round $i \leq N$, student i is selected, and a_i is greedily replaced by the sequence s^* that minimizes total gain, or simply restated, provides the largest drop in the average gain from A. Formally,

$$s^* = \arg\min_{s \in P_{SQ}} g((s, a_{-i})) \tag{9}$$

$$= \arg\min_{s \in P_{SQ}} g((s, a_{-i})) - g(A) \tag{10}$$

where (s, a_{-i}) denotes the assignment where student i's sequence a_i is replaced with s. Note that for any $s \in P_{SQ}$, the difference in the average gain between (s, a_{-i}) and A is the sum of the differences in the gain from each question j that appears in the sequence s, as shown in Eq. (11).

$$g((s, a_{-i})) - g(A) = \frac{1}{N} \sum_{j \in s} \Big[\sum_{k > i} p_{k,j} [y_k \beta(a_k, j, s(j)) + y_i (1 - \beta(a_k, j, s(j)))] - [y_k \delta(a_i, a_k, j) + y_i (1 - \delta(a_i, a_k, j))] + \sum_{h > i} p_{i,h} [y_i \gamma(a_i, j, s(j)) + y_h (1 - \gamma(a_i, j, s(j)))] [y_i \delta(a_h, a_i, j) + y_h (1 - \delta(a_h, a_i, j))] \Big] \tag{11}$$

We compute $s^* = \arg\min_{s \in P_{SQ}} g((s, a_{-i}))$ by solving the following minimum weight maximum matching problem to match questions to positions in a sequence. We define a weighted, complete, bipartite graph $G = ([M_1] \cup [M_2], E)$ with a node for each of M_1 positions, and a node for each of M_2 questions. For each pair of a position $l \in [M_1]$ and question $j \in [M_2]$, we set the weight of the edge to be the difference in the gain from question j when it appears in position l and the gain from question j as it appears in a_i, w.r.t. the sequences a_{-i} of all of the other students. It is easy to see that solving this minimum weight maximum matching problem assigns student i with a desired sequence of M_1 questions $s^* = \arg\min_{s \in P_{SQ}} g((s, a_{-i})) - g(A) = \arg\min_{s \in P_{SQ}} g((s, a_{-i}))$.

Then, we extended MMM into the MMM-CGS algorithm as a natural extension of MMM and CGS by setting the initial assignment to the output of CGS (modifying line 2 in Algorithm 3 in the Supplementary Note 2) and improving it greedily in the same manner as MMM. This ensures that we will only improve solutions from the CGS (at least no harm), which implies a room for potential improvement of our heuristic optimization method CGS.

Integer linear programming (ILP)

For the optimal performance, we adapted this setting into an integer linear programming problem to find an optimal assignment in the permutation space but at an exponential computational cost, as shown in Algorithm 4 in Supplementary Note 2.

We begin by showing correctness of Algorithm 4, and that it computes a valid solution. Consider an arbitrary instance $I = ([N], M_2, M_1, Y)$, and let A be the assignment returned by Algorithm 4 when applied on this instance I. It is easy to see that for any student $i \in [N]$, (ii) for any question $j \in [M_2]$, there is at most one value of $l \in [M_1]$, such that $s_{i,j} = l$, otherwise the constraint $\sum_{j \in [M_1]} m_{i,j,l} = 1$ is violated, and (ii) for any position $l \in [M_1]$, there is exactly one question $j \in [M_2]$ such that $s_{i,j} = l$, otherwise, together with (i), the constraint $\sum_{j \in [M_1]} s_{i,j} = \sum_{l \in [M_2]} l$ is violated. It is easy to see by the construction of Algorithm 4, that every student is assigned M_1 questions in A in a valid sequence.

It is easy to verify that the objective of the ILP formulation in Algorithm 4 is the score of the assignment indicated by the variables $s_{i,j}$, by checking that for each pair of students $i, k \in [N]$, and for each question $j \in [M_2]$, the variables $c_{i,k,j}$ correctly indicate whether i can copy from k on question j under the assignment indicated by the variables $s_{i,j}$ and $s_{k,j}$.

To prove completeness, it is sufficient to verify that every possible assignment is a feasible solution to the ILP in Algorithm 4. It is easy to check that for every valid assignment A, there is a way to assign values first to variables $s_{i,j}$ corresponding to the sequences in A, and subsequently to the rest of the variables in the ILP formulation in a manner that does not violate any of the constraints.

Practical guideline

The GAS itself is usually not optimal, but using its result as the initialization of the greedy algorithms can guarantee the theoretical bound of the average collusion gain of the searching results. In our simulation, the performance of our fast heuristic search algorithm CGS was close to the results optimized using the two sophisticated algorithms MMM and ILP (Supplementary Note 3). Note that our CGS method does not guarantee convergence on the optimum, and is theoretically different from the competitive sophisticated algorithms. Especially, the ILP algorithm finds the global optimum but requires exponentially more computational resources. To design online exams of small scales, we generally prefer using ILP as appropriate. For online exams of large scales, we generally prefer using the MMM algorithm that is of polynomial complexity to find at least a local minimum, initialized by the output of our GAS and CGS methods.

Data collection and ethics oversight

We developed a DOT platform (a web application, all MCQs, detailed in Supplementary Note 4 and 6) to perform the online test. The post-exam survey was performed through SurveyMonkey. We have complied with all relevant ethical regulations. The RPI Institutional Review Board approved the study protocol. The informed consent was obtained from all participants in the study.

Reporting summary

Further information on research design is available in the Nature Research Reporting Summary linked to this article.

DATA AVAILABILITY

We declare that all data that support the findings of this study are already available within the manuscript and the supplementary information. Raw data of the tests will be made available upon request and after going through an Institutional Review Board procedure at RPI.

CODE AVAILABILITY

The source codes and simulation data for this study are publicly available on Github (https://github.com/edward1001001/Anticheating_in_DOT).

Received: 8 July 2020; Accepted: 7 December 2020;
Published online: 01 March 2021

REFERENCES

1. Roediger, H. L., III, Putnam, A. L., & Smith, M. A. Ten benefits of testing and their applications to educational practice. In: J. P. Mestre & B. H. Ross (Eds) The psychology of learning and motivation: Cognition in education. San Diego, CA: Elsevier Academic Press, vol. 55, p1–36 (2011).
2. Diekhoff, G. M. et al. College cheating: ten years later. *Res. High. Educ.* **37**, 487–502 (1996).
3. Galante, M. The 10 biggest college cheating scandals. *Business Insider* (2012).
4. McCabe, D. L., Butterfield, K. D. & Trevino, L. K. *Cheating in College: Why Students Do It and What Educators Can Do About It* (JHU Press, 2012).
5. Toquero, C. M. Challenges and opportunities for higher education amid the COVID-19 pandemic: the Philippine context. *Pedagog. Res.* **5**, em0063 (2020).
6. Vlachopoulos, D. Covid-19: threat or opportunity for online education? *High. Learn. Res. Commun.* **10**, 2 (2020).
7. Hodges, C., Moore, S., Lockee, B., Trust, T. & Bond, A. The difference between emergency remote teaching and online learning. *Educause Review*. https://er.educause.edu/articles/2020/3/the-difference-between-emergency-remote-teaching-and-online-learning (2020).
8. Adnan, M. & Anwar, K. Online learning amid the COVID-19 pandemic: Students' perspectives. *J. Pedagog. Soc. Psychol.* **2**, 45–51 (2020).

9. Zaharah, Z., Kirilova, G. I. & Windarti, A. Impact of corona virus outbreak towards teaching and learning activities in Indonesia. *SALAM J. Sos. dan Budaya Syar-i* **7**, 269–282 (2020).
10. Mailizar, M., Almanthari, A., Maulina, S. & Bruce, S. Secondary school mathematics teachers' views on e-learning implementation barriers during the COVID-19 pandemic: the case of Indonesia. *Eurasia J. Math. Sci. Technol. Educ.* **16**, em1860 (2020).
11. McPherson, M. S. & Bacow, L. S. Online higher education: neyond the hype cycle. *J. Econ. Perspect.* **29**, 135–54 (2015).
12. Mustafa, N. Impact of the 2019–2020 coronavirus pandemic on education. *Int. J. Health Preferences Res.* **5**, 31 (2020).
13. Chen, B., Azad, S., Fowler, M., West, M. & Zilles, C. Learning to cheat: quantifying changes in score advantage of unproctored assessments over time. In *Proceedings of the Seventh ACM Conference on Learning@ Scale*, Virtual Event, USA, 12–14 August 2020, pp. 197–206 (2020).
14. Fuller, R., Joynes, V., Cooper, J., Boursicot, K. & Roberts, T. Could COVID-19 be our 'there is no alternative'(TINA) opportunity to enhance assessment? *Med. Teach.* **42**, 781–786 (2020).
15. Rowe, N. C. Cheating in online student assessment: beyond plagiarism. *Online J. Distance Learn. Admin.* **7**, https://www.westga.edu/~distance/ojdla/summer72/rowe72.html (2004).
16. Alessio, H. M., Malay, N., Maurer, K., Bailer, A. J. & Rubin, B. Examining the effect of proctoring on online test scores. *Online Learn.* **21**, 146–161 (2017).
17. Watson, G. R. & Sottile, J. Cheating in the digital age: do students cheat more in online courses? *Online J. Distance Learn. Admin.* **13**, https://www.westga.edu/~distance/ojdla/spring131/watson131.html (2010).
18. Lee, J., Kim, R. J., Park, S.-Y. & Henning, M. A. Using technologies to prevent cheating in remote assessments during the COVID-19 pandemic. *J. Dent. Educ.* 1–3. https://doi.org/10.1002/jdd.12350 (2020).
19. Farmer, J., Lachter, G. D., Blaustein, J. J. & Cole, B. K. The role of proctoring in personalized instruction1. *J. Appl. Behav. Anal.* **5**, 401–404 (1972).
20. Medina, M. S. & Castleberry, A. N. Proctoring strategies for computer-based and paper-based tests. *Am. J. Health-System Pharmacy* **73**, 274–277 (2016).
21. TOP HAT. Online proctoring software. https://tophat.com/online-proctoring/ (2020).
22. Examity. https://examity.com/ (2020).
23. Proctortrack. https://www.proctortrack.com (2020).
24. Harwell, D. Mass school closures in the wake of the coronavirus are driving a new wave of student surveillance. *Washington Post* 1 (2020).
25. Lilley, M., Meere, J. & Barker, T. Remote live invigilation: a pilot study. *J. Interactive Media Educ.* **6**, 1–5 (2016).
26. Sullivan, D. P. An integrated approach to preempt cheating on asynchronous, objective, online assessments in graduate business classes. *Online Learning* **20**, 195–209 (2016).
27. Chin, M. Exam anxiety: how remote test-proctoring is creeping students out. *The Verge* 29 (2020).
28. Amigud, A., Arnedo-Moreno, J., Daradoumis, T. & Guerrero-Roldan, A.-E. Open proctor: an academic integrity tool for the open learning environment. In: Barolli L., Woungang I., Hussain O. K. (Eds) *The 9th International Conference on Intelligent Networking and Collaborative Systems* Ryerson University, Canada, August 24–26 2017 (Lecture Notes on Data Engineering and Communications Technologies) vol **8**, Springer, Heidlberg, p262–273 (2017).
29. Amigud, A., Arnedo-Moreno, J., Daradoumis, T. & Guerrero-Roldan, A.-E. Using learning analytics for preserving academic integrity. *Int. Rev. Res. Open Distr. Learn. IRRODL* **18**, 192–210 (2017).
30. Trenholm, S. A review of cheating in fully asynchronous online courses: a math or fact-based course perspective. *J. Educ. Technol. Syst.* **35**, 281–300 (2007).
31. Cluskey Jr, G., Ehlen, C. R. & Raiborn, M. H. Thwarting online exam cheating without proctor supervision. *J. Acad. Bus. Ethics* **4**, 1–7 (2011).
32. Golden, J. & Kohlbeck, M. Addressing cheating when using test bank questions in online classes. *J. Account. Educ.* **52**, 100671 (2020).
33. Mazar, N., Amir, O. & Ariely, D. The dishonesty of honest people: a theory of self-concept maintenance. *J. Market. Res.* **45**, 633–644 (2008).
34. Connolly, J., Lentz, P. & Morrison, J. Using the business fraud triangle to predict academic dishonesty among business students. *AELJ* **10**, 37 (2006).
35. Madara, B. et al. Nursing students' access to test banks: Are your tests secure? *J. Nurs. Educ.* **56**, 292–294 (2017).
36. Burns, C. M. Sold! web-based auction sites have just compromised your test bank. *Nurse Educ.* **34**, 95–96 (2009).
37. Cheng, C. & Crumbley, D. L. Student and professor use of publisher test banks and implications for fair play. *J. Account. Educ.* **42**, 1–16 (2018).
38. Corrigan-Gibbs, H., Gupta, N., Northcutt, C., Cutrell, E. & Thies, W. Deterring cheating in online environments. *ACM Trans. Comput. Hum. Interact.* **22**, 1–23 (2015).
39. Golub, E. Pcs in the classroom & open book exams. *Ubiquity* **2005**, 1–1 (2005).
40. Nelson, G. E. On-line evaluation: multiple choice, discussion questions, essay, and authentic projects. In *Third Annual Teaching in the Community College Conference* (Kapiolani Community College, Honolulu, Hawaii, 1998).
41. Sam, A. H., Reid, M. D. & Amin, A. High-stakes remote-access open-book examinations. *Med. Educ.* **54**, 767–768 (2020).
42. Ullah, A., Xiao, H. & Lilley, M. Profile based student authentication in online examination. In *International Conference on Information Society (i-Society 2012)*, London, UK, 25–28 June 2012, p 109–113. https://ieeexplore.ieee.org/abstract/document/6285058 (2012).
43. Bailie, J. L. & Jortberg, M. A. Online learner authentication: verifying the identity of online users. *J. Online Learn. Teach.* **5**, 197–207 (2009).
44. Ullah, A., Xiao, H., Lilley, M. & Barker, T. Using challenge questions for student authentication in online examination. *Int. J. Infonomics* **5**, 9 (2012).
45. Haynie, W. Effects of multiple-choice and short-answer tests on delayed retention learning. *J. Technol. Educ.* **6**, 32–44 (1994).
46. Abdel-Hameed, A. A., Al-Faris, E. A., Alorainy, I. A. & Al-Rukban, M. O. The criteria and analysis of good multiple choice questions in a health professional setting. *Saudi Med. J.* **26**, 1505–1510 (2005).
47. Anderson, T. W. & Darling, D. A. Asymptotic theory of certain "goodness of fit" criteria based on stochastic processes. *Ann. Math. Stat* **23**, 193–212 (1952).
48. Dodge, Y. *Kolmogorov–Smirnov Test*. In: The Concise Encyclopedia of Statistics. 283–287 (Springer, New York, 2008).
49. Montgomery, D. C. & Runger, G. C. *Applied Statistics and Probability for Engineers* 4th edn (Wiley, 2007).
50. Wilcox, R. R. *Introduction to Robust Estimation and Hypothesis Testing* (Academic Press, 2011).
51. Underwood, S. Blockchain beyond bitcoin. *Commun. ACM* **59**, 15–17 (2016).
52. Wang, R., Hozumi, Y., Yin, C. & Wei, G.-W. Decoding SARS-CoV-2 tTransmission and evolution and ramifications for COVID-19 diagnosis, vaccine, and medicine. *J. Chem. Inf. Model.* **60**, 5853–5865 (2020).
53. Kissler, S. M., Tedijanto, C., Goldstein, E., Grad, Y. H. & Lipsitch, M. Projecting the transmission dynamics of SARS-CoV-2 through the postpandemic period. *Science* **368**, 860–868 (2020).

AUTHOR CONTRIBUTIONS

Conceptualization: G.W., M.L., H.M., and L.X.; methodology: M.L., S.S., L.X., U.K., and G.W.; software and Web: L.L. and H.S.; data curation: N.I.N., S.G., and M.L.; statistical analysis: M.K. and U.K.; project administration: G.W., H.M., and L.X.; writing and editing: all.

COMPETING INTERESTS

G.W., M.L., L.X., and S.S. have a pending patent application on the presented technology.

ADDITIONAL INFORMATION

Supplementary information Supplementary information is available for this paper at https://doi.org/10.1038/s41539-020-00083-3.

Correspondence and requests for materials should be addressed to G.W.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.