Graph Design for Secure Multiparty Computation over Non-Abelian Groups

Xiaoming Sun[1], Andrew Chi-Chih Yao[1], and Christophe Tartary[1,2]

[1] Institute for Theoretical Computer Science
Tsinghua University
Beijing, 100084
People's Republic of China

[2] Division of Mathematical Sciences
School of Physical and Mathematical Sciences
Nanyang Technological University
Singapore

{xiaomings,andrewcyao}@tsinghua.edu.cn
ctartary@ntu.edu.sg

Abstract. Recently, Desmedt et al. studied the problem of achieving secure n-party computation over non-Abelian groups. They considered the passive adversary model and they assumed that the parties were only allowed to perform black-box operations over the finite group G. They showed three results for the n-product function $f_G(x_1, \ldots, x_n) := x_1 \cdot x_2 \cdot \ldots \cdot x_n$, where the input of party P_i is $x_i \in G$ for $i \in \{1, \ldots, n\}$. First, if $t \geq \lceil \frac{n}{2} \rceil$ then it is impossible to have a t-private protocol computing f_G. Second, they demonstrated that one could t-privately compute f_G for any $t \leq \lceil \frac{n}{2} \rceil - 1$ in exponential communication cost. Third, they constructed a randomized algorithm with $O(n\,t^2)$ communication complexity for any $t < \frac{n}{2.948}$.
In this paper, we extend these results in two directions. First, we use percolation theory to show that for any fixed $\epsilon > 0$, one can design a randomized algorithm for any $t \leq \frac{n}{2+\epsilon}$ using $O(n^3)$ communication complexity, thus nearly matching the known upper bound $\lceil \frac{n}{2} \rceil - 1$. This is the first time that percolation theory is used for multiparty computation. Second, we exhibit a deterministic construction having polynomial communication cost for any $t = O(n^{1-\epsilon})$ (again for any fixed $\epsilon > 0$). Our results extend to the more general function $\widetilde{f}_G(x_1, \ldots, x_m) := x_1 \cdot x_2 \cdot \ldots \cdot x_m$ where $m \geq n$ and each of the n parties holds one or more input values.

Keywords: Multiparty Computation, Passive Adversary, Non-Abelian Groups, Graph Coloring, Percolation Theory.

1 Introduction

In multiparty computation, a set of n parties $\{P_1, \ldots, P_n\}$ want to compute a function of some secret inputs held locally by these participants. Since its introduction by Yao

[19], multiparty computation has been extensively studied. Most multiparty computation protocols rely on algebraic structures which are at least Abelian groups [14] as in [1, 3, 4, 8, 10, 11, 12] for instance. The usefulness of Abelian groups in cryptography is not restricted to multiparty computation as numerous cryptographic primitives are developed over such groups [6, 7, 17]. However, the construction of efficient quantum algorithms to solve the discrete logarithm problem as well as the factoring problem prevent the use of many of these primitives over those machines [18]. Since quantum algorithms seem to be less efficient over non-Abelian groups, there is increasingly a need for developing cryptographic constructions over such mathematical structures. The reader may be aware of the existence of public key cryptosystems for such groups [15, 16].

Recently, Desmedt et al. studied the problem of designing secure n-party protocol over non commutative finite groups for the *passive* (or *semi-honest*) adversary model [5]. Their goal is to guarantee unconditional security simply using a black-box representation of the finite non-Abelian group $(G, \cdot)$. This assumption means that the n parties can only perform three operations in $(G, \cdot)$: the group operation $((x, y) \mapsto x \cdot y)$, the group inversion $(x \mapsto x^{-1})$ and the uniformly distributed group sampling $(x \in_R G)$.

Desmedt et al. focused on the existence and the design of t-private protocols for the n product function $f_G(x_1, \ldots, x_n) := x_1 \cdot \ldots \cdot x_n$ where the input of party P_i is $x_i \in G$ for $i \in \{1, \ldots, n\}$. In such a protocol, no colluding sets $\mathcal{C}$ of at most t participants learn anything about the data hold by any of the remaining members $\{P_1, \ldots, P_n\} \setminus \mathcal{C}$. Desmedt et al. obtained three important results. First, if $t \geq \lceil \frac{n}{2} \rceil$ (dishonest majority) then it is impossible to construct a t-private protocol to compute f_G. Second, if $t < \lceil \frac{n}{2} \rceil$ then one can always design a deterministic t-private protocol computing f_G with an exponential communication complexity of $O(n \binom{2\,t+1}{t}^2)$ group elements. Third, they built a probabilistic t-private protocol computing f_G with a polynomial communication complexity of $O(n\,t^2)$ group elements when $t < \frac{n}{2.948}$.

That work leads to two important questions. First, we would like to know if it is possible to construct a t-private protocol for values of $t \in \left[\frac{n}{2.948}, \lceil \frac{n}{2} \rceil - 1\right]$ with polynomial communication complexity. Second, Desmedt et al.'s construction shows that one can t-privately compute f_G with polynomial communication cost for any $t = O(\log n)$. A natural issue is to determine the existence and to construct a deterministic t-private protocol with polynomial communication complexity for other values t (ideally, up to the threshold $\lceil \frac{n}{2} \rceil - 1$).

In this article, we give a positive answer to these two questions. First, we demonstrate that the random coloring approach and the graph construction by Desmedt et al. can be used to guarantee t-privacy for any $t < \frac{n}{2+\epsilon}$ (for any fixed $\epsilon > 0$). The communication complexity of our construction is $O(n^3)$ group elements. This result is obtained using percolation theory. To the best of our knowledge, this is the first use of this theory in the context of multiparty computation. Second, we provide a deterministic construction for any $t = O(n^{1-\epsilon})$. This scheme has polynomial communication complexity as

well.

This paper is organized as follows. In the next section, we will recall the different reductions performed in [5] to solve the t-privacy issue over non-Abelian groups. In Sect. 3, we present our randomized construction achieving t-privacy for any value $t \leq \frac{n}{2+\epsilon}$ which is closed to the theoretical bound $\lceil \frac{n}{2} \rceil - 1$. In Sect. 4, we show how to construct deterministic t-private protocols having polynomial communication cost for any $t = O(n^{1-\epsilon})$. In the last section, we conclude our paper with some remaining open problems for multiparty computation over non-Abelian black-box groups.

2 Achieving Secure Computation over Non-Abelian Groups

In this section, we present some of the results and constructions developed by Desmedt et al. which are necessary to understand our improvements from Sect. 3 and Sect. 4. First, we recall the definition of secure multiparty computation in the passive, computationally unbounded attack model, restricted to deterministic symmetric functionalities and perfect emulation as in [5].

We denote $[n]$ the set of integers $\{1, \ldots, n\}$, $\{0,1\}^*$ the set of all finite binary strings and $|A|$ the cardinality of the set A.

Definition 1. *We denote $f : (\{0,1\}^*)^n \mapsto \{0,1\}^*$ an n-input and single-output function. Let $\prod$ be a n-party protocol for computing f. We denote the n-party input sequence by $\mathbf{x} = (x_1, \ldots, x_n)$, the joint protocol view of parties in subset $I \subset [n]$ by $\mathbf{VIEW}_I^{\prod}(\mathbf{x})$, and the protocol output by $\mathbf{OUT}^{\prod}(\mathbf{x})$. For $0 < t < n$, we say that $\prod$ is a t-private protocol for computing f if there exists a probabilistic polynomial-time algorithm S, such that, for every $I \subset [n]$ with $|I| \leq t$ and every $\mathbf{x} \in (\{0,1\}^*)^n$, the random variables*

$$\langle S(I, \mathbf{x}_I, f(\mathbf{x})), f(\mathbf{x}) \rangle \text{ and } \langle \mathbf{VIEW}_I^{\prod}(\mathbf{x}), \mathbf{OUT}^{\prod}(\mathbf{x}) \rangle$$

are identically distributed, where $\mathbf{x}_I$ denotes the projection of the n-ary sequence $\mathbf{x}$ on the coordinates in I.

In the remaining of this paper, we assume that party P_i has a personal input $x_i \in G$ (for $i \in [n]$) and the function to be computed is the n-party product $f_G(x_1, \ldots, x_n) := x_1 \cdot \ldots \cdot x_n$.

Desmedt et al. first reduced the problem of constructing a t-private n-party protocol for f_G to the problem of constructing a *symmetric (strong) t-private protocol* $\prod'$ (see [5] for a detailed definition of symmetric privacy) to compute the shared 2-product function $f'_G(x, y) := x \cdot y$ where the inputs x and y are shared amongst the n parties. They demonstrated that iterating $(n-1)$ times the protocol $\prod'$ would give a t-private protocol to compute f_G.

The second reduction occurring in [5] consists of constructing a t-private n-party shared 2-product protocol $\prod'$ from a suitable coloring over particular directed graphs. We will detail the important steps of this reduction as they will serve the understanding of our own constructions.

Definition 2 ([5]). *We call graph $\mathcal{G}$ an* admissible Planar Directed Acyclic Graph (PDAG) *with share parameter ℓ and size parameter $m(\geq \ell)$ if it has the following properties:*

- *The nodes of $\mathcal{G}$ are drawn on a square $m \times m$ grid of points (each node of $\mathcal{G}$ is located at a grid point but some grid points may not be occupied by nodes). The rows of the grid are indexed from top to bottom and the columns from left to right by the integers $1, 2, \ldots, m$. A node of $\mathcal{G}$ at row i and column j is said to have index (i, j). $\mathcal{G}$ has $2\,\ell$ input nodes on the top row, and ℓ output nodes on the bottom row.*
- *The incoming edges of a node on row i only come from nodes on row $i-1$, and outgoing edges of a node on row i only go to nodes on row $i+1$.*
- *For each row i and column j, let $\eta_1^{(i,j)} < \cdots < \eta_{q^{(i,j)}}^{(i,j)}$ denote the ordered column indices of the $q(i,j) > 0$ nodes on level $i+1$ which are connected to node (i,j) by an edge. Then, for each $j \in [m-1]$, we have:*

$$\eta_{q(i,j)}^{(i,j)} \leq \eta_1^{(i,j+1)}$$

which means that the rightmost node on level $i+1$ connected to node (i,j) is to the left of (or equal to) the leftmost node on level $i+1$ connected to node $(i, j+1)$.

An admissible PDAG has 2ℓ input nodes. The first ℓ ones (i.e. $(1,1), \ldots, (1,\ell)$) represent the x-input nodes while the remaining ones represent the y-input nodes. Let $C : [m] \times [m] \mapsto [n]$ be a n-coloring function that associates to each node (i,j) of $\mathcal{G}$ a color $C(i,j)$ chosen from a set of n possible colors. The following notion will be used to express the property we expect the graph coloring to have in order to build $\prod'$.

Definition 3 ([5]). *We say that $C : [m] \times [m] \mapsto [n]$ is a t-*reliable *n-*coloring *for the admissible* PDAG *$\mathcal{G}$ (with share parameter ℓ and size parameter m) if for each t-color subset $I \subset [n]$, there exist $j^* \in [\ell]$ and $j_y^* \in [\ell]$ such that:*

- *There exists a path* PATH_x *in $\mathcal{G}$ from the j^*th x-input node to the j^*th output node, such that none of the path node colors are in subset I (it is called an I-avoiding path), and*
- *There exists an I-avoiding path* PATH_y *in $\mathcal{G}$ from the j_y^*th y-input node to the j^*th output node.*

If $j_y^* = j^*$ for all I, we say that C is a symmetric *t-reliable n-coloring.*

Important Remark: Even if the graph $\mathcal{G}$ is directed, it is regarded as *non-directed* when building the I-avoiding paths in Definition 3.

Desmedt et al. built a protocol $\prod'(\mathcal{G}, C)$ taking as input a graph $\mathcal{G}$ and a n coloring C. We do not detail this protocol in our paper as its internal design does not have

any influence in our work. The reader can find it in [5]. However, in order to ease the understanding of our work, we recall the relation between multiparty protocols over a non-Abelian group G and coloring of admissible PDAGs as it appear in [5].

The n participants $\{P_1, \ldots, P_n\}$ are identified by the n colors of the admissible PDAG $\mathcal{G}$. The input/output nodes of the graph $\mathcal{G}$ are labeled by the input/output elements of the group G. Each edge represents a group element sent from one participant to another one. Each internal node contains an intermediate value of the protocol. Those values are computed, at each node $\mathcal{N}$ of $\mathcal{G}$, as the group operation between the elements along all the incoming edges of $\mathcal{N}$ from the leftmost one to the rightmost one. This intermediate value is then redistributed along all the outgoing edges of $\mathcal{N}$ using the following $O_{\mathcal{N}}$-of-$O_{\mathcal{N}}$ secret sharing where $O_{\mathcal{N}}$ represents the number of outgoing edges of node $\mathcal{N}$.

Proposition 1 ([5]). *Let g be an element of the non-Abelian group G. Denote λ and μ two integers where $\mu \in [\lambda]$. We create a λ-of-λ sharing $(s_g(1), \ldots, s_g(\lambda))$ of g by picking the $\lambda - 1$ shares $\{s_g(\xi)\}_{\xi \in [\lambda] \setminus \{\mu\}}$ uniformly and independently at random from G, and computing $s_g(\mu)$ to be the unique element of G such that:*

$$g = s_g(1) \cdot s_g(2) \cdot \ldots \cdot s_g(\lambda)$$

Then, the distribution of the shares $(s_g(1), \ldots, s_g(\lambda))$ is independent of μ.

We recall the following important result:

Theorem 1 ([5]). *If $\mathcal{G}$ is an admissible* PDAG *and C is a symmetric t-reliable n-coloring for $\mathcal{G}$ then $\prod'(\mathcal{G}, C)$ achieves symmetric strong t-privacy.*

The last reduction is related to the admissible PDAG. Desmedt et al. only consider admissible PDAGs as defined below and represented in Fig. 1.

Definition 4 ([5]). *The admissible* PDAG *$\mathcal{G}_{tri}(\ell', \ell)$ is a $\ell' \times \ell$ directed grid such that:*

- *[horizontal edges] for $i \in [\ell']$ and for $j \in [\ell - 1]$, there is a directed edge from node $(i, j+1)$ to (i, j),*
- *[vertical edges] for $i \in [\ell' - 1]$ and for $j \in [\ell]$, there is a directed edge from node (i, j) to node $(i+1, j)$,*
- *[diagonal edges] for $i \in [\ell' - 1]$ and for $j \in \{2, \ldots, \ell\}$, there is a directed edge from node (i, j) to node $(i+1, j-1)$.*

According to Definition 2, an admissible PDAG has $2\,\ell$ input nodes and no horizontal edges. Desmedt et al. indicated that the y-input nodes could be arranged along a column on $\mathcal{G}_{tri}(\ell', \ell)$ instead of being along the same row as the x-input nodes. They also explained that $\mathcal{G}_{tri}(\ell', \ell)$ could also be drawn according the requirements of Definition 2. By rotating $\mathcal{G}_{tri}(\ell', \ell)$ by 45 degrees anticlockwise, the x-input nodes and y-input nodes of $\mathcal{G}_{tri}(\ell', \ell)$ are now on the same row and the horizontal edges of $\mathcal{G}_{tri}(\ell', \ell)$ have become diagonal edges which satisfies Definition 2.

A priori, $\mathcal{G}_{tri}(\ell', \ell)$ is a rectangular grid. In [5], Desmedt et al. considered square grids $\mathcal{G}_{tri}(\ell, \ell)$ for which they introduced the following notion.

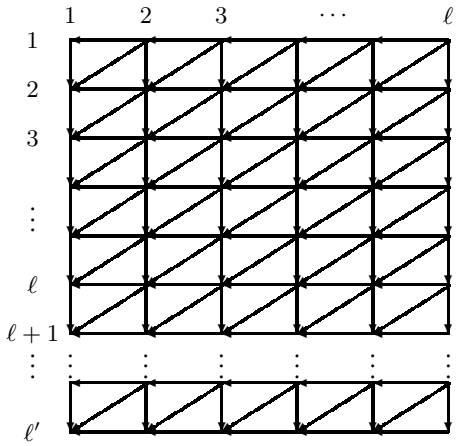

Fig. 1. The admissible PDAG $\mathcal{G}_{tri}(\ell', \ell)$.

Definition 5 ([5]). *We say that $C : [\ell] \times [\ell] \mapsto [n]$ is a weakly t-reliable n-coloring for $\mathcal{G}_{tri}(\ell, \ell)$ if for each t-color subset $I \subset [n]$:*

- *There exists an I-avoiding path $\mathcal{P}_x$ in $\mathcal{G}_{tri}(\ell, \ell)$ from a node on the top row to a node on the bottom row. Such a path is called an I-avoiding top-bottom path.*
- *There exists an I-avoiding path $\mathcal{P}_y$ in $\mathcal{G}_{tri}(\ell, \ell)$ from a node on the rightmost column to a node on the leftmost column. Such a path is called an I-avoiding right-left path.*

As said in [5], the admissible PDAG requirements (Definition 2) are still satisfied if we remove from $\mathcal{G}_{tri}$ some 'positive slope' diagonal edges and add some 'negative slope' diagonal edges (connecting a node (i, j) to node $(i+1, j+1)$, for some $i \in [\ell'-1]$ and $j \in [\ell - 1]$). Such a generalized admissible PDAG is denoted $\mathcal{G}_{gtri}$.

Lemma 1 ([5]). *Let $C : [\ell] \times [\ell] \mapsto [n]$ be a weakly t-reliable n-coloring for square admissible* PDAG $\mathcal{G}_{tri}(\ell, \ell)$. *Then, we can construct a t-reliable n-coloring for a rectangular admissible* PDAG $\mathcal{G}_{gtri}(2\ell - 1, \ell)$.

Thus, Desmedt et al. have demonstrated that it was sufficient to get a weakly t-reliable n coloring for some $\mathcal{G}_{tri}(\ell, \ell)$ in order to construct a t-private protocol for computing the n-product f_G. The cost communication cost of this protocol is $(n - 1)$ times the number of edges of $\mathcal{G}_{gtri}(2\ell-1, \ell)$. Since that grid is obtained from $\mathcal{G}_{tri}(\ell, \ell)$ using a mirror, the communication cost of the whole protocol is $O(n\,\ell^2)$ group elements. The constructions that we propose in this paper are colorings of some grids $\mathcal{G}_{tri}(\ell, \ell)$.

3 A Randomized Construction Achieving Maximal Privacy

In this section, we present a randomized construction ensuring the t-privacy of the computation of f_G up to $\frac{n}{2+\epsilon}$. Our scheme has a linear share parameter $\ell = O(n)$.

We use the same random coloring C_{rand} for the grid $\mathcal{G}_{tri}(\ell, \ell)$ as in [5]. However, our analysis is based on percolation theory while Desmedt et al. used a counting-based argument. We first introduce the following definition which is illustrated in Fig. 2.

Algorithm 1 Coloring C_{rand}

Input: A grid $\mathcal{G}_{tri}(\ell, \ell)$.

1. For each $(i, j) \in [\ell] \times [\ell]$, choose the color $C(i, j)$ of node (i, j) independently and uniformly at random from $[n]$.

Output: A n-coloring of the grid.

Definition 6. *The* triangular lattice of depth ℓ *denoted* $\mathcal{T}(\ell)$ *is a directed graph drawn over a* $\ell \times (3\,\ell - 2)$ *grid such that:*

- *[horizontal edges] for* $i \in [\ell]$ *and for* $j \in [\ell - 1]$*, there is a directed edge from node* $(i, i + 2\,j)$ *to* $(i, i + 2\,(j - 1))$*,*
- *[right downwards edges] for* $i \in [\ell - 1]$ *and for* $j \in \{0, \ldots, \ell - 1\}$*, there is a directed edge from node* $(i, i + 2\,j)$ *to node* $(i + 1, i + 2\,j + 1)$*,*
- *[left downwards edges] for* $i \in [\ell - 1]$ *and for* $j \in [\ell - 1]$*, there is a directed edge from node* $(i, i + 2\,j)$ *to node* $(i + 1, i + 2\,j - 1)$*.*

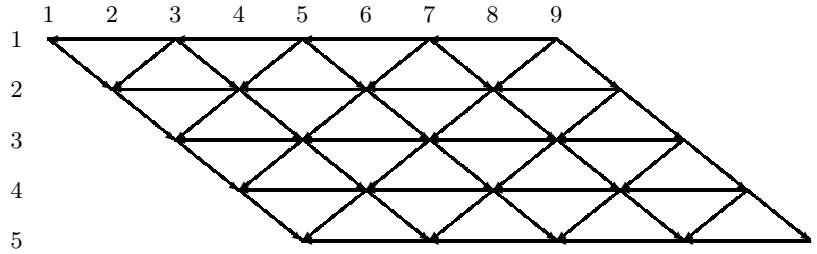

Fig. 2. The triangle $\mathcal{T}(5)$.

Proposition 2. *For any positive integer* ℓ*, we have a graph isomorphism between* $\mathcal{G}_{tri}(\ell, \ell)$ *and* $\mathcal{T}(\ell)$*.*

Proof. Consider the mapping:

$$\begin{aligned} \mathcal{G}_{tri}(\ell, \ell) &\longrightarrow \mathcal{T}(\ell) \\ (i, j) &\longmapsto (i, i + 2\,(j - 1)) \end{aligned}$$

It is easy to see that the nodes of the two graphs are in bijective correspondence while the direction of each edge is maintained. □

Theorem 2. *For any $\epsilon > 0$, there exists a constant c_ϵ such that if $t \leq \frac{n}{2+\epsilon}$ and $\ell \geq c_\epsilon n$, then there exists a weakly t-reliable n-coloring for $\mathcal{G}_{tri}(\ell, \ell)$.*

Proof. We prove that the coloring C_{rand} will work with high probability. Let $t_\epsilon = \left\lfloor \frac{n}{2+\epsilon} \right\rfloor$ where $\lfloor \cdot \rfloor$ denotes the floor function. Instead of considering the probability that C_{rand} is a weakly t_ϵ-reliable n-coloring for $\mathcal{G}_{tri}(\ell, \ell)$, we study the complementary event. A suitable value for ℓ will be given at the end of this demonstration.

The coloring C_{rand} is called *bad* if there exists a color set $I \subset [n]$ with $|I| = t_\epsilon$, such that either there are no I-avoiding top-bottom paths or there are no I-avoiding right-left paths. By the union bound, we obtain the following upper bound on $\Pr(C_{rand}$ is bad$)$:

$$\begin{aligned} &2 \Pr(\exists I \subset [n], |I| = t_\epsilon, \text{there are no } I\text{-avoiding top-bottom paths in } \mathcal{G}_{tri}(\ell, \ell)) \\ &\leq 2 \sum_{I \subset [n], |I| = t_\epsilon} \Pr(\text{there are no } I\text{-avoiding top-bottom paths in } \mathcal{G}_{tri}(\ell, \ell)). \end{aligned} \tag{1}$$

The factor 2 in (1) comes from the fact the top-bottom probability is equal to the right-left probability due to the symmetry of the grid $\mathcal{G}_{tri}(\ell, \ell)$ and the coloring C_{rand}.

Next, we demonstrate that for a fixed color set $I \subset [n]$ with $|I| = t_\epsilon$, the probability that there are no I-avoiding top-bottom paths in C_{rand} is exponentially small. Let us fix the color set I. We call a vertex *closed* if its color belongs to I. Otherwise, the vertex is called *open*. The random coloring C_{rand} of each vertex is equivalent to open it independently and randomly with probability $p := 1 - \frac{t_\epsilon}{n}$. An I-avoiding path is simply an *open path*. Therefore, we get:

$$\begin{aligned} &\Pr(\text{there are no } I\text{-avoiding top-bottom paths in } \mathcal{G}_{tri}(\ell, \ell)) \\ &= \Pr_p(\text{there are no open top-bottom paths in } \mathcal{G}_{tri}(\ell, \ell)) \\ &= 1 - \Pr_p(\text{there is an open top-bottom path in } \mathcal{G}_{tri}(\ell, \ell)) \end{aligned} \tag{2}$$

We have the following result.

Lemma 2 ([2]). *The triangular lattice $\mathcal{T}(\ell)$ has the following property:*

$$\begin{gathered} \Pr_p(\textit{there is an open top-bottom path in } \mathcal{T}(\ell)) \\ + \\ \Pr_p(\textit{there is a closed right-left path in } \mathcal{T}(\ell)) \\ = 1 \end{gathered}$$

When we combine Lemma 2, Proposition 2 and (2), we obtain the following:

$$\begin{gathered} \Pr(\text{there is no } I\text{-avoiding top-bottom path in } \mathcal{G}_{tri}(\ell, \ell)) \\ = \\ \Pr_p(\text{there is a closed right-left path in } \mathcal{T}(\ell)) \\ = \\ \Pr_{1-p}(\text{there is an open right-left path in } \mathcal{T}(\ell)) \end{gathered} \tag{3}$$

In (3), $\Pr_{1-p}(\cdot)$ means that we open each vertex with probability $1-p$. We have the following result from percolation theory.

Lemma 3 ([13]). *Let T be the triangular lattice in the plane. Then, the critical probability of site percolation $p_c^s(T)$ is equal to $\frac{1}{2}$.*

When the open probability is less than the critical probability, the percolation has the following properties (see for example Chapter 4, Theorem 9 in [2]).

Lemma 4 ([9]). *If $p < p_c^s(T)$, then there is a constant $c = c(p)$,*

$$\Pr_p(0 \xrightarrow{n}) < e^{-c\,n}.$$

where $\{x \xrightarrow{n}\}$ is the event that there is an open path from x to a point in $S_n(x)$ with $S_n(x) := \{y : d(x,y) = n\}$ and $d(x,y)$ denotes the distance between x and y.

Remark: The value 0 from Lemma 4 represent the zero element of $\mathbb{Z} \times \mathbb{Z}$ when the graph is represented as a lattice over that set. In the case of the triangular lattice depicted as Fig. 2, the value 0 can be identified to the node $(1,1)$.

In our case, we have: $1 - p = \frac{t_\epsilon}{n} \leq \frac{1}{2+\epsilon} < p_c^s(T)$. Using Lemma 4, we get:

$$\Pr_{1-p}(\text{there is an open right-left path in } \mathcal{T}(\ell)) \leq \ell \Pr_{1-p}(0 \xrightarrow{\ell-1}) \leq \ell\, e^{-c\,(\ell-1)} \quad (4)$$

The first inequality is due to the fact that any right-left path has length at least $(\ell - 1)$ in $\mathcal{T}(\ell)$. Combining (1)-(4), we obtain:

$$\Pr(C_{rand} \text{ is bad}) \leq 2 \binom{n}{t_\epsilon} \ell\, e^{-c\,(\ell-1)}$$

Thus, if we choose $\ell := c_\epsilon\, n$ for some large enough constant c_ϵ, we have:

$$\Pr(C_{rand} \text{ is bad}) \leq \frac{1}{2^n}$$

which guarantees the fact that C_{rand} is a weakly t_ϵ-reliable n-coloring for $\mathcal{G}_{tri}(\ell, \ell)$ with overwhelming probability in n. □

Corollary 1. *There exists a black box t_ϵ-private protocol for f_G with communication complexity $O(n^3)$ group elements where $t_\epsilon = \lfloor \frac{n}{2+\epsilon} \rfloor$. Moreover, for any $\delta > 0$, we can construct a probabilistic algorithm, with run-time polynomial in n and $\log(\delta^{-1})$, which outputs a protocol $\prod$ for f_G such that the communication complexity of $\prod$ is $O(n^3 \log^2(\delta^{-1}))$ group elements and the probability that $\prod$ is not t_ϵ-private is at most δ.*

Proof. The existence of the protocol is a direct consequence of Theorem 2 as well as the different reductions exposed in Sect. 2. As our construction requires $\ell = O(n)$, we deduce that the communication cost of the protocol computing f_G is $O(n^3)$. The justification of the running time of the algorithm and the probability of failure δ is identical to what is done in [5]. □

We showed that it was possible to build a randomized algorithm to achieve $\left\lfloor \frac{n}{2+\epsilon} \right\rfloor$-private computation of f_G using $O(n^3)$ group elements. Even if the probability of failure of our previous construction is small, we would like to remove the randomized restriction so that we can get a (deterministic) protocol which is always guaranteed to succeed. In [5], Desmedt et al. only provided deterministic protocols to compute f_G in polynomial communication cost when $t = O(\log n)$. In the next section, we present a deterministic construction for any $t = O(n^{1-\epsilon})$ where ϵ is any positive constant. Our construction requires polynomial communication complexity as well.

4 A Deterministic Construction for Secure Computation

In this section, we show how to build a deterministic t-private protocol to compute f_G with polynomial complexity cost for any $t = O(n^{1-\epsilon})$. First, we will focus on particular pairs (t, n). Second, we generalize our result to any (t, n) with $t = O(n^{1-\epsilon})$.

We recursively construct our admissible PDAG $\mathcal{G}_{rec}$ and its coloring C_{rec}. Let $d \in \mathbb{N} \setminus \{0, 1\}$ be a constant. Denote $\mathcal{B}_d$ the binomial coefficient $\binom{2d-1}{d-1}$.

Theorem 3. *For any positive integer k, there is a weakly t_k-reliable n_k-coloring $C_{rec}(\ell_k)$ for the square admissible* PDAG *$\mathcal{G}_{rec}(\ell_k)$, where the parameters are: $t_k := d^k - 1$, $n_k := (2d-1)^k$ and $\ell_k = \mathcal{B}_d^k \left(\mathcal{B}_d + 1\right)^{k-1}$.*

Proof. We prove the theorem by induction on k.

$k = 1$: We have $t_1 = d - 1$, $n_1 = 2\,d - 1$ and $\ell_1 = \mathcal{B}_d$. We set $\mathcal{G}_{rec}(\ell_1) := \mathcal{G}_{tri}(\ell_1, \ell_1)$. We define $C_{rec}(\ell_1)$ as being the combinatorial coloring C_{comb} designed in [5] and recalled as Algorithm 2.

Algorithm 2 Coloring C_{comb}

Input: A $L \times L$ grid where $L = \binom{N}{T}$.

1. Let $I_1, \ldots, I_L$ denote the sequence of all T-color subsets of $[N]$ (in some ordering).
2. For each $(i, j) \in [L] \times [L]$, define the color $C(i, j)$ of node (i, j) in the grid to be any color in the set $S_{i,j} := [N] \setminus (I_i \cup I_j)$.

Output: A N-coloring of the grid.

Desmedt et al. noticed that, even if we removed the diagonal edges from $\mathcal{G}_{tri}(\ell_1, \ell_1)$, we still had the existence of I-avoiding top-bottom and right-left paths. Thus, we assume that $\mathcal{G}_{rec}(\ell_1)$ has no such edges so that $\mathcal{G}_{rec}(\ell_1)$ is a square grid the side length of which is ℓ_1 nodes. $\mathcal{G}_{rec}(\ell_1)$ is an admissible PDAG.

$k \geq 1$: Suppose we already have the construction and coloring for k, we recursively construct $\mathcal{G}_{rec}(\ell_{k+1})$ from $\mathcal{G}_{rec}(\ell_k)$.

We first build the block grid B by copying $(\mathcal{B}_d + 1) \times (\mathcal{B}_d + 1)$ times $\mathcal{G}_{rec}(\ell_1)$. The connections between two copies of $\mathcal{G}_{rec}(\ell_1)$ are as follows. Horizontally, we draw a directed edge from node $(i, 1)$ in the right-hand side copy to node (i, ℓ_1) in the left-hand side copy for $i \in [\ell_1]$ (i.e. we horizontally connect nodes at the same level). Vertically, we draw a directed edge from node (ℓ_1, j) in the top side copy to node $(1, j)$ in the bottom side copy for $j \in [\ell_1]$ (i.e. we vertically connect nodes at the same level).

The block B is a $(\mathcal{B}_d\,(\mathcal{B}_d + 1)) \times (\mathcal{B}_d\,(\mathcal{B}_d + 1))$ grid. It has the following property the proof of which can be found in Appendix A.

Proposition 3. *The block grid B admits a $(2\,d - 1)$-coloring (just use the same C_{comb} for each copy of $\mathcal{G}_{rec}(\ell_1)$), such that for any $(d - 1)$-color subset $I \subset [2\,d - 1]$, there are $\mathcal{B}_d + 1$ horizontal (vertical) I-avoiding* **straight lines** *in B.*

Now, we construct $\mathcal{G}_{rec}(\ell_{k+1})$ and its coloring $C_{rec}(\ell_{k+1})$ as follows. We replace each node in $\mathcal{G}_{rec}(\ell_k)$ by a copy of B. If the node of $\mathcal{G}_{rec}(\ell_k)$ was colored by the color $c \in [n_k]$, then we color B with the set of colors $\{(2d-1)(c-1)+1, (2d-1)(c-1)+2, \ldots, (2\,d-1)\,c\}$, using C_{comb}. All the edges within each copy of B remain identical in $\mathcal{G}_{rec}(\ell_{k+1})$.

Now, we show how to connect two copies of B. We first focus on vertical connections. Consider an edge in $\mathcal{G}_{rec}(\ell_k)$ from a node in the i-th row to another node in the $(i + 1)$-th row. Since these two nodes have been replaced by two copies of B, we denote the nodes on the top copy (i.e. those corresponding to the nodes of the i-th row in $\mathcal{G}_{rec}(\ell_k)$) as $v_{1,1}, \ldots, v_{1,\mathcal{B}_d}, v_{2,1}, \ldots, v_{\mathcal{B}_d+1,\mathcal{B}_d}$ and the nodes on the bottom copy as $w_{1,1}, \ldots, w_{1,\mathcal{B}_d}, w_{2,1}, \ldots, w_{\mathcal{B}_d+1,\mathcal{B}_d}$.

For each $(i, j) \in [\mathcal{B}_d] \times [\mathcal{B}_d]$, we add a directed edge $(v_{i,j}, w_{i,j+i-1})$ in $\mathcal{G}_{rec}(\ell_{k+1})$. If the index $(j + i - 1)$ is greater than $\mathcal{B}_d$, $w_{i,j+i-1}$ is the node $w_{i+1,j+i-1-\mathcal{B}_d}$. Figure 3 gives the example for $d = 2$. The connection process works similarly for two consecutive columns where we replace each horizontal edge from $\mathcal{G}_{rec}(\ell_k)$ by $\mathcal{B}_d^2$ different edges in $\mathcal{G}_{rec}(\ell_{k+1})$.

It is clear that the number of nodes on each side of the square $\mathcal{G}_{rec}(\ell_{k+1})$ is:

$$\ell_{k+1} = \mathcal{B}_d\,(\mathcal{B}_d + 1) \cdot \ell_k = \mathcal{B}_d^{k+1}\,(\mathcal{B}_d + 1)^k$$

and the number of colors used in $C_{rec}(\ell_{k+1})$ is $n_{k+1} = (2\,d - 1) \cdot n_k = (2\,d - 1)^{k+1}$. The grid $\mathcal{G}_{rec}(\ell_{k+1})$ obtained by this recursive process is also an admissible PDAG due to the horizontal/vertical connection processes between two copies of B (as well as two copies of $\mathcal{G}_{rec}(\ell_1)$ inside B).

The last point to prove is that for any t_{k+1}-color subset $I \subset [n_{k+1}]$, there is an I-avoiding top-bottom (and right-left) path in $\mathcal{G}_{rec}(\ell_{k+1})$. We only prove the existence of a top-bottom path in this paper as the demonstration of the existence for a right-left path is similar. For each $j \in [n_k]$, we define the set I_j as:

$$I_j := I \cap \{(2d-1)(j-1)+1, (2d-1)(j-1)+2, \ldots, (2\,d-1)\,j\}$$

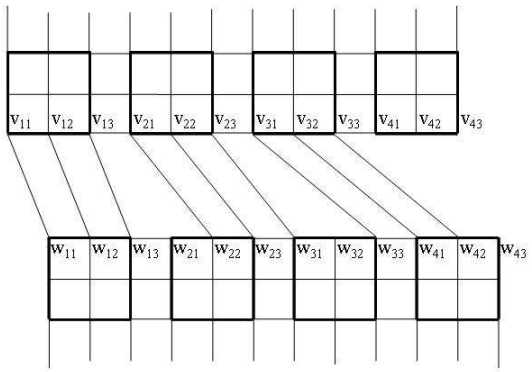

Fig. 3. How to vertically connect two copies of B when $d = 2$.

Since

$$|I_1| + \cdots + |I_{n_k}| = |I| = t_{k+1} = d^{k+1} - 1 \tag{5}$$

and each $|I_j| \leq 2d - 1$, there are at least $(n_k - t_k)$ subsets having at most $(d - 1)$ elements. Indeed, in the opposite case, we would have:

$$|I_1| + \cdots + |I_{n_k}| \geq d\,(n_k - (n_k - t_k - 1)) = d \cdot d^k = d^{k+1},$$

which would contradict (5). Assume that $S \subseteq [n_k]$ is the set of these indices (i.e. for each $j \in S$, $|I_j| \leq d - 1$). We have: $|[n_k] \setminus S| \leq t_k$. By the induction hypothesis, there is a $([n_k] \setminus S)$-avoiding top-bottom path in $\mathcal{G}_{rec}(\ell_k)$, i.e., the colors used on this path all belong to S. Let $v_1, \ldots, v_m$ be the vertices of the path and denote the color of node v_j as $c_j \in S$ $(j \in [m])$.

Now, we show there is an I-avoiding top-bottom path in $\mathcal{G}_{rec}(\ell_{k+1})$. In $\mathcal{G}_{rec}(\ell_{k+1})$, each node v_j has been replaced by a copy B_{v_j} with colors in $\{(2d - 1)(c_j - 1) + 1, (2d - 1)(c_j - 1) + 2, \ldots, (2\,d - 1)\,c_j\}$. Since the color set I_{c_j} satisfies $|I_{c_j}| \leq d - 1$, by Proposition 3 we deduce that there are $\mathcal{B}_d$ horizontal and $\mathcal{B}_d$ vertical I_{c_j}-avoiding paths in B_{v_j}.

One can show that this property involves the existence of an I-avoiding top-bottom path in $\mathcal{G}_{rec}(\ell_{k+1})$. This top-bottom path is the connection of an I_{c_1}-avoiding path (from B_{v_1}), an I_{c_2}-avoiding path (from B_{v_2}),. . ., an I_{c_m}-avoiding path (from B_{v_m}). The reader can find more details about this process in Appendix B. A similar demonstration leads to the existence of an I-avoiding right-left path in $\mathcal{G}_{rec}(\ell_{k+1})$ which achieves the demonstration of our theorem. □

The communication complexity of the protocol to t_k-privately compute the function $f_G(x_1, \ldots, x_{n_k})$ using the previous admissible PDAG is $O(n_k\,\ell_k^2)$ group elements where:

$$\ell_k \leq \mathcal{B}_d^k(\mathcal{B}_d + 1)^{k-1} \leq 2^{(2d-1)k} \times 2^{(2d-1)(k-1)} \leq 2^{2k(2d-1)} \leq n_k^{\frac{2(2d-1)}{\log_2(2d-1)}}$$

Note that the last inequality comes from $2^k = n_k^{\frac{1}{\log_2(2d-1)}}$.

Now, we generalize our result to any (t, n) where $t = O(n^{1-\epsilon})$ for any fixed positive ϵ. The class $O(n^{1-\epsilon})$ is the set of all functions f such that: $\exists \tau_f > 0 \, \exists n_0 > 0 : \forall n \geq n_0 \, f(n) \leq \tau_f \, n^{1-\epsilon}$. In our case, the function f is the privacy level t. Our main result is stated as follows.

Theorem 4. *For any fixed $\epsilon > 0$, for any fixed $\tau > 0$, there exists a constant $n_{\epsilon,\tau} \in \mathbb{N}$, such that for any $n \geq n_{\epsilon,\tau}$, if $t \leq \tau \, n^{1-\epsilon}$, then there exists a black-box t-private protocol to compute f_G with communication complexity polynomial in n. Moreover, there is a deterministic polynomial time algorithm to construct the protocol.*

Proof. We fix $\epsilon > 0$ and $\tau > 0$. We set $d = 2^{\lceil \frac{2}{\epsilon} \rceil - 1}$ and $k = \lfloor \log_{(2d-1)} n \rfloor$. We have $d \geq 2$. If $n \geq 2\,d - 1$ then $k \geq 1$. In such a condition, we can apply Theorem 3 for the pair (k, d). There exists a t_k-private protocol to compute the value $f_G(x_1, \ldots, x_{n_k})$ using $O(n_k \, \ell_k^2)$ group elements where t_k, n_k, ℓ_k are defined as in Theorem 3. It is clear that the construction also t'-privately computes $f_G(x_1, \ldots, x_{n'})$ for any (t', n') such that $t' \leq t_k$ and $n' \geq n_k$. So, we only need to show $\tau \, n^{1-\epsilon} \leq t_k$, $n \geq n_k$ and $\ell_k = \text{poly}(n)$. Due to our choice of d and k, we have:

$$n_k \leq (2d-1)^{\lfloor \log_{(2d-1)} n \rfloor} \leq (2d-1)^{\log_{(2d-1)} n} \leq n$$

And:

$$t_k \geq d^{\lfloor \log_{(2d-1)} n \rfloor} - 1 \geq d^{\log_{(2d-1)} n - 1} - 1 \geq \frac{n^{\frac{\log_2 d}{\log_2 (2d-1)}}}{d} - 1 \geq \frac{n^{\frac{\log_2 d}{\log_2 2d}}}{d} - 1$$

Since $d = 2^{\lceil \frac{2}{\epsilon} \rceil - 1}$, we get:

$$t_k \geq \frac{n^{\frac{\lceil \frac{2}{\epsilon} \rceil - 1}{\lceil \frac{2}{\epsilon} \rceil}}}{2^{\lceil \frac{2}{\epsilon} \rceil - 1}} - 1 \geq \frac{n^{1-\frac{\epsilon}{2}}}{2^{\lceil \frac{2}{\epsilon} \rceil - 1}} - 1 \geq \frac{n^{\frac{\epsilon}{2}}}{2^{\lceil \frac{2}{\epsilon} \rceil - 1}} \, n^{1-\epsilon} - 1$$

Since ϵ is a fixed positive constant, the mapping $n \mapsto \frac{n^{\frac{\epsilon}{2}}}{2^{\lceil \frac{2}{\epsilon} \rceil - 1}}$ has an infinite limit. Therefore: $\exists \widetilde{n}_{\epsilon,\tau} > 0 : \forall n \geq \widetilde{n}_{\epsilon,\tau} \quad \frac{n^{\frac{\epsilon}{2}}}{2^{\lceil \frac{2}{\epsilon} \rceil - 1}} \geq \tau + \frac{1}{n^{1-\epsilon}}$.

Remember that we early required $n \geq 2\,d - 1$ in order to use Theorem 3. If we set $n_{\epsilon,\tau} := \max(2\,d - 1, \widetilde{n}_{\epsilon,\tau})$ then:

$$\forall n \geq n_{\epsilon,\tau} \begin{cases} n_k \leq n \\ t_k \geq \tau \, n^{1-\epsilon} \geq t \end{cases}$$

It remains to argue about ℓ_k. Since $n_k \leq n$, we have: $\ell_k \leq n^{\frac{2\,(2\,d-1)}{\log_2\,(2\,d-1)}}$. Since d is independent from n, ℓ_k is upper bounded by a polynomial in n. □

The previous theorem claims that for any fixed ϵ, if n is chosen large enough then we can t-privately compute f_G for any $t = O(n^{1-\epsilon})$. Such an asymptotic survey is also

performed in [5]. However, in practical applications, the number of participants is not asymptotically large. The deterministic construction by Desmedt et al. has polynomial cost when $t = O(\log n)$. We now present a result valid for any group size n which guarantees privacy for larger t's than in [5] using polynomial communication as well.

Theorem 5. *For any positive integer n no smaller than 3, there exists a black-box protocol for f_G which is $(\lceil \frac{n^{\log_3 2}}{2} \rceil - 1)$-private. It requires the n participants to exchange $O(n^6)$ group elements. Moreover, there is a deterministic polynomial time algorithm to construct the protocol.*

Proof. We set $d = 2$ and $k := \lfloor \log_3(n) \rfloor$. The protocol obtained using Theorem 3 has parameter $t_k \geq \frac{n^{\log_3 2}}{2} - 1$ and $n_k \leq n$. We have: $\mathcal{B}_2 = 3$. Therefore: $\ell_k \leq \frac{n^{1+2\log_3 2}}{4}$. Thus, we obtain: $n_k \ell_k^2 = O(n^6)$. □

5 Conclusion and Open Problems

In this paper, we first demonstrated that we could construct a probabilistic t-private protocol computing the n-product function over any non-Abelian group for *any* t up to $\frac{n}{2+\epsilon}$ (for any fixed positive ϵ), thus nearly matching the known upper bound $\lceil \frac{n}{2} \rceil - 1$. As the communication complexity of our construction is $O(n^3)$ group elements, this result answers one of the questions asked by Desmedt et al. concerning the largest collision resistance achievable with an admissible PDAG of size polynomial in n. Note that Desmedt et al. indicated the discovery of a construction for $(n, t) = (24, 11)$ improving locally their own theoretical bound $\frac{n}{2.948}$ since $11 \approx \frac{24}{2.182}$. Our result demonstrates the existence of such a construction for any fixed positive ϵ (in [5], we have the particular case $\epsilon = 0.182$). Since the scheme developed in [5] (exclusively valid for $t < \frac{n}{2.948}$) only requires $O(n\,t^2)$ elements to be exchanged, a direction to further investigate is the existence of a (randomized) t-private protocol for any $t \leq \lceil \frac{n}{2} \rceil - 1$ having at most the cost of Desmedt et al.'s scheme.

Second, we showed that it was possible to construct a deterministic t-private n-party protocol to compute f_G having a polynomial communication cost for any $t = O(n^{1-\epsilon})$. For practical purpose, one may want to optimize the choice of parameters in our construction. For example, we have proved that one could t-privately compute f_G for any (t, n) satisfying $t \leq \left\lceil \frac{n^{\log_3 2}}{2} \right\rceil - 1$.

Desmedt et al. argued that the reduction from a protocol computing the n-product to a subroutine computing the shared 2-product extended to the more general function $\widetilde{f}_G(x_1, \ldots, x_m) := x_1 \cdot x_2 \cdot \ldots \cdot x_m$ where $m \geq n$ and each of the n parties holds one or more input values. This ensured the validity of their protocol to securely compute $\widetilde{f}_G$ as well. Since the constructions that we presented are particular admissible PDAGs, our results are also valid to compute $\widetilde{f}_G$.

Our work leads to the following two questions. First, is it possible to reduce the communication cost when $t = O(n^{1-\epsilon})$? Second, can we generalize this approach to

design a deterministic polynomial communication cost algorithm for any t up to the threshold $\lceil \frac{n}{2} \rceil - 1$?

Apart from the previous points which constitute directions to improve the security for the passive adversary model, a problem which requires attention is the possibility of achieving secure computation of f_G against malicious parties. Indeed, even if multiparty computation can be used with small groups (as in the case of the Millionaires' problem [19]), the general purpose is to enable large communication groups to perform common computations and the larger the number of parties is, the more likely (at least) one of them will deviate from the given protocol.

Acknowledgments

The authors are grateful to the anonymous reviewers for their comments to improve the quality of this paper. The three authors' work was supported in part by the National Natural Science Foundation of China grant 60553001 and the National Basic Research Program of China grants 2007CB807900 and 2007CB807901. Xiaoming Sun's research was also funded by the National Natural Science Foundation of China under grant 60603005. Christophe Tartary's work was also financed by the Ministry of Education of Singapore under grant T206B2204.

References

[1] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *20th Annual ACM Symposium on Theory of Computing*, pages 1 – 10, Chicago, USA, May 1988. ACM Press.

[2] B. Bollobàs and O. Riordan. *Percolation*. Cambridge University Press, September 2006.

[3] R. Cramer, I. B. Damgård, and U. Maurer. General secure multi-party computation from any linear secret-sharing scheme. In *Advances in Cryptology - Eurocrypt '00*, volume 1807 of *Lecture Notes in Computer Science*, pages 316 – 334, Bruges, Belgium, May 2000. Springer - Verlag.

[4] I. B. Damgård and Y. Ishai. Scalable secure multiparty computation. In *Advances in Cryptology - Crypto '06*, volume 4117 of *Lecture Notes in Computer Science*, pages 501 – 520, Santa Barbara, USA, August 2006. Springer.

[5] Y. Desmedt, J. Pieprzyk, R. Steinfeld, and H. Wang. On secure multi-party computation in black-box groups. In *Advances in Cryptology - Crypto '07*, volume 4622 of *Lecture Notes in Computer Science*, pages 591 – 612, Santa Barbara, USA, August 2007. Springer - Verlag.

[6] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644 – 654, November 1976.

[7] T. El Gamal. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469 – 472, 1985.

[8] O. Goldreich and R. Vainish. How to solve any protocol problem - an efficiency improvement. In *Advances in Cryptology - Crypto '87*, volume 293 of *Lecture Notes in Computer Science*, pages 73 – 86, Santa Barbara, USA, August 1988. Springer - Verlag.

[9] J. M. Hammersley. Percolation processes: Lower bounds for the critical probability. *The Annals of Mathematical Statistics*, 28(3):790 – 795, September 1957.

[10] M. Hirt and U. Maurer. Robustness for free in unconditional multi-party computation. In *Advances in Cryptology - Crypto '01*, volume 2139 of *Lecture Notes in Computer Sciences*, pages 101 – 118, Santa Barbara, USA, August 2001. Springer - Verlag.
[11] M. Hirt, U. Maurer, and B. Przydatek. Efficient secure multi-party computation. In *Advances in Cryptology - Asiacrypt '00*, volume 1976 of *Lecture Notes in Computer Science*, pages 143 – 161, Kyoto, Japan, December 2000. Springer - Verlag.
[12] M. Hirt and J. B. Nielsen. Robust multiparty computation with linear communication complexity. In *Advances in Cryptology - Crypto' 06*, volume 4117 of *Lecture Notes in Computer Science*, pages 463 – 482, Santa Barbara, USA, August 2006. Springer.
[13] H. Kesten. *Percolation Theory for Mathematicians*. Birkhäuser, November 1982.
[14] S. Lang. *Algebra (Revised Third Edition)*. Springer, November 2002.
[15] S. S. Magliveras, D. R. Stinson, and T. van Trung. New approaches to designing public key cryptosystems using one-way functions and trapdoors in finite groups. *Journal of Cryptology*, 15(4):285 – 297, 2002.
[16] S.-H. Paeng, K.-C. Ha, J. H. Kim, S. Chee, and C. Park. New public key cryptosystem using finite non Abelian groups. In *Advances in Cryptology - Crypto '01*, volume 2139 of *Lecture Notes in Computer Science*, pages 470 – 485, Santa Barbara, USA, August 2001. Springer - Verlag.
[17] R. L. Rivest, A. Shamir, and L. M. Adleman. A method for obtaining digital signatures and public key cryptosystems. *Communication of the ACM*, 21(2):120 – 126, February 1978.
[18] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484 – 1509, 1997.
[19] A. C.-C. Yao. Protocols for secure computations. In *23rd Annual IEEE Symposium on Foundations of Computer Science*, pages 80 – 91, Chicago, USA, November 1982. IEEE Press.

A Proof of Proposition 3

Let I be a $(d-1)$-color subset of $[2\,d-1]$. In [5], Desmedt et al. demonstrated that there were a I-avoiding top-bottom path and a I-avoiding right-left path in $\mathcal{G}_{tri}(\ell_1,\ell_1)$. They also showed that those two paths were straight lines. Thus, one can remove the diagonal edges of $\mathcal{G}_{tri}(\ell_1,\ell_1)$ while preserving those paths. This means that there exist a I-avoiding top-bottom path and a I-avoiding right-left path in $\mathcal{G}_{rec}(\ell_1)$ which are straight lines.

Since B is a-$(\mathcal{B}_d+1)\times(\mathcal{B}_d+1)$-copy of $\mathcal{G}_{rec}(\ell_1)$ and, due to the vertical/horizontal connections of these copies, we deduce that there are $(\mathcal{B}_d+1)$ I-avoiding top-bottom paths and $(\mathcal{B}_d+1)$ I-avoiding right-left paths in B. Moreover, each of these paths is a straight line.

B Connection of Color Avoiding Paths

It was shown in the proof of Theorem 3 that each block B_{c_i} had $\mathcal{B}_d$ horizontal and $\mathcal{B}_d$ vertical I_{c_i}-avoiding paths. In this appendix, we show how to construct a I-avoiding top-bottom path in $\mathcal{G}_{rec}(\ell_{k+1})$. Our path will start at the top of B_{v_1} and ends at the bottom of B_{v_m}.

Every grid from the family $(\mathcal{G}_{rec}(\ell_\lambda))_{\lambda\geq 1}$ is a square grid. Thus, the sequence of blocks $B_{v_1},\ldots,B_{v_m}$ in $\mathcal{G}_{rec}(\ell_{k+1})$ is determined by the position of B_{v_1} as well as the m-tuple of letters from $\{\mathfrak{L},\mathfrak{R},\mathfrak{T},\mathfrak{B}\}$ ($\mathfrak{L}$eft, $\mathfrak{R}$ight, $\mathfrak{T}$op, $\mathfrak{B}$ottom) indicating the output side of the block B_{v_i} for $i\in[m]$. Note that the last letter of the tuple is always $\mathfrak{B}$ since the I-avoiding top-bottom path ends at the bottom of B_{v_m}.

This tuple has the property the two consecutive letters cannot be opposite to each other (i.e, one cannot have $(\mathfrak{L},\mathfrak{R})$, $(\mathfrak{R},\mathfrak{L})$, $(\mathfrak{T},\mathfrak{B})$ or $(\mathfrak{B},\mathfrak{T})$). This means that you leave a block on a different side that you entered it. The reader can check the correctness of this claim by a simple recursive process on the parameter k. This property is trivially true for $k=1$ since $\mathcal{G}_{rec}(\ell_1)=\mathcal{G}_{tri}(\ell_1)$. The recursion follows from the path construction that we will design below.

Proposition 4. *Let i be any element of $[m]$. Assume that $\mathcal{N}$ is any node on a side of B_{v_i} belonging to a I_{c_i}-avoiding straight line path. For each other side $\mathfrak{S}_i$ of B_{v_i}, we can construct a I_{c_i}-avoiding path from $\mathcal{N}$ to any of the $(\mathcal{B}_d+1)$ nodes on $\mathfrak{S}_i$ belonging to a I_{c_i}-avoiding straight line path.*

Proof. We only provide a proof when $\mathcal{N}$ is on the top side of B_{v_i} (the three other cases are similar). The three possible output sides are $\mathfrak{B},\mathfrak{L}$ and $\mathfrak{R}$. The block B_{v_i} is a-$(\mathcal{B}_d+1)\times(\mathcal{B}_d+1)$-copy of the original grid $\mathcal{G}_{rec}(\ell_1)$. Thus, B_{v_i} can be treated as a $(\mathcal{B}_d+1)\times(\mathcal{B}_d+1)$ array of grids $\mathcal{G}_{rec}(\ell_1)$. Based on this observation, we will use the terminology *grid-row* (respectively *grid-column*) to denote a set of $\mathcal{B}_d+1$ horizontal (respectively vertical) grids $\mathcal{G}_{rec}(\ell_1)$ in B_{v_i}.

1. $\underline{\mathfrak{S}_i=\mathfrak{B}}$. The vertical I_{c_i}-avoiding path starting at node $\mathcal{N}$ intersects the **horizontal** I_{c_i}-avoiding path located within the bottom grid-row of B_{v_i} at node $\mathcal{I}$. That horizontal path intersects each of the $\mathcal{B}_d+1$ **vertical** I_{c_i}-avoiding paths (one within each grid-column) at $\mathcal{I}_1,\ldots,\mathcal{I}_{\mathcal{B}_d+1}$. Note that $\mathcal{I}=\mathcal{I}_\mu$ for some $\mu\in[\mathcal{B}_d+1]$. Once we are at one of the $\mathcal{I}_j$'s, we simply go vertically downwards to the node $\mathcal{N}'_j$ located at the bottom side of the block B_{v_i}.

Thus, we can construct a path from $\mathcal{N}$ to each of the $\mathcal{B}_d+1$ output nodes on the bottom side of B_{v_j} belonging to the vertical I_{c_i}-avoiding paths. Those paths are $(\mathcal{N},\mathcal{I},\mathcal{I}_j,\mathcal{N}'_j)$ for $j\in[\mathcal{B}_d+1]$.

2. $\underline{\mathfrak{S}_i=\mathfrak{R}}$. The vertical I_{c_i}-avoiding path starting at node $\mathcal{N}$ intersects the **horizontal** I_{c_i}-avoiding path located within the top grid-row of B_{v_i} at node $\mathcal{I}$. That horizontal path intersects the **vertical** I_{c_i}-avoiding path located within the rightmost grid-column of B_{v_i} at node $\widetilde{\mathcal{I}}$. This vertical path intersects each of the $\mathcal{B}_d+1$ **horizontal** I_{c_i}-avoiding paths (one within each grid-row) at $\widetilde{\mathcal{I}}_1,\ldots,\widetilde{\mathcal{I}}_{\mathcal{B}_d+1}$. As before, we get: $\widetilde{\mathcal{I}}=\widetilde{\mathcal{I}}_\mu$ for some $\mu\in[\mathcal{B}_d+1]$. Once we are at one of the $\widetilde{\mathcal{I}}_j$'s, we horizontally go rightwards to the node $\mathcal{N}'_j$ located on the right hand side of the block B_{v_i}.

Thus, we can construct a path from $\mathcal{N}$ to each of the $\mathcal{B}_d+1$ output nodes on the right hand side of B_{v_j} belonging to the horizontal I_{c_i}-avoiding paths. Those paths are

$(\mathcal{N}, \mathcal{I}, \widetilde{\mathcal{I}}, \widetilde{\mathcal{I}}_j, \mathcal{N}'_j)$ for $j \in [\mathcal{B}_d + 1]$.

3. $\underline{\mathfrak{S}_i = \mathfrak{L}}$. This is analogous to the previous case. □

We can finally construct a I-avoiding top-bottom path in $\mathcal{G}_{rec}(\ell_{k+1})$. We denote the m-tuple of output sides as $(\mathfrak{S}_1, \ldots, \mathfrak{S}_m)$. As previously said, we have: $\mathfrak{S}_m = \mathfrak{B}$.

We start at **any** node N_1 located on the top side of B_{v_1} and on a vertical I_{c_1}-avoiding path. Using Proposition 4, we can connect N_1 to any of the $\mathcal{B}_d + 1$ nodes on side $\mathfrak{S}_1$ of B_{v_1} using a I_{c_1}-avoiding path. An important remark is that each block of the whole grid $\mathcal{G}_{rec}(\ell_{k+1})$ is a set of $(\mathcal{B}_d + 1) \times (\mathcal{B}_d + 1)$ identical copies of $\mathcal{G}_{rec}(\ell_1)$ (including the coloring). As a consequence, these $\mathcal{B}_d + 1$ nodes have the same location in their respective copies of $\mathcal{G}_{rec}(\ell_1)$. Given the connection process between any pair of blocks within $\mathcal{G}_{rec}(\ell_{k+1})$, one of these $\mathcal{B}_d + 1$ nodes must be connected to a node N_2 from block B_{v_2} belonging to a I_{c_2}-avoiding straight line path. Similarly, N_2 is connected via a I_{c_2}-avoiding path in B_{v_2} to a node N_3 from B_{v_3} belonging to a I_{c_3}-avoiding straight line path. If we repeat this process for each of the remaining blocks, we obtain a set of $m - 1$ nodes $N_1, \ldots, N_{m-1}$. The last node N_{m-1} can be connected to a node N_m on the bottom side of B_{v_m} using a I_{c_m}-avoiding path. Thus, N_1 (top side of $\mathcal{G}_{rec}(\ell_{k+1})$) is connected to N_m (bottom side of $\mathcal{G}_{rec}(\ell_{k+1})$) using a I-avoiding path which achieves the demonstration of our theorem.

Remark: As claimed above, this construction involves that the two consecutive side letters of the m-tuple cannot be opposite to each other.