entropy

IMPACT FACTOR 2.1

Indexed in: PubMed

CITESCORE 4.9

Article

# Communication Complexity of Entanglement-Assisted Multi-Party Computation

Ruoyu Meng and Aditya Ramamoorthy

Special Issue

Entropy, Quantum Information and Entanglement

Edited by

Dr. Paul M. Alsing

MDPI

https://doi.org/10.3390/e26110896

entropy

MDPI

*Article*

# Communication Complexity of Entanglement-Assisted Multi-Party Computation

**Ruoyu Meng * and Aditya Ramamoorthy**

Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA; adityar@iastate.edu

* Correspondence: rmeng@iastate.edu

**Abstract:** We consider a quantum and a classical version of a multi-party function computation problem with $n$ players, where players $2, \ldots, n$ need to communicate appropriate information to player 1 so that a "generalized" inner product function with an appropriate promise can be calculated. In the quantum version of the protocol, the players have access to entangled qudits but the communication is still classical. The communication complexity of a given protocol is the total number of classical bits that need to be communicated. When $n$ is prime, and for our chosen function, we exhibit a quantum protocol (with complexity $(n-1)(\log n)$ bits) and a classical protocol (with complexity $((n-1)^2(\log n^2)$ bits)). Furthermore, we present an integer linear programming formulation for determining a lower bound on the classical communication complexity. This demonstrates that our quantum protocol is strictly better than classical protocols.

**Keywords:** communication complexity; entanglement-assisted communication; integer linear programming

## 1. Introduction

We consider a multi-party function computation scenario in this work. There are a total of $n$ players in the system numbered $1, 2, \ldots, n$. Each player observes her input and players $2, \ldots, n$ (remote parties) communicate an appropriate number of bits that allows player 1 to finally compute the value of the function. Clearly, this can be accomplished if players $2, \ldots, n$ communicate their actual values, but in many cases, the function value can be computed with much less information. Thus, a key question is to determine the minimum number of bits the remote parties need to send to player 1.

Such problems are broadly studied under the umbrella of communication complexity [1,2] in the literature. In this work, we consider the zero-error version of this problem. Our main goal is to understand the advantage that the availability of quantum entanglement confers on this problem and compare it with classical protocols. Such problems have a long history in the literature [3,4].

**Background:** There are three kinds of quantum protocols within quantum communication complexity (QCC) problems. In the first kind (introduced by Yao [2]), each player communicates via a quantum channel and the metric is the number of qubits transmitted. We call it the quantum transmission model. The second variation assumes that each player can use entanglement as a free resource, but the communication remains classical; the primary metric here is the number of classical bits transmitted. We refer to this as the entanglement model, which was introduced by Cleve and Buhrman [5]. The third kind is a combination of the first two. We call it the combined model. It allows free usage of entanglement and works with quantum communication. The work of de Wolf [6] shows that, in the two-party case, the entanglement model can be reduced to the quantum transmission model with a two-fold penalty, utilizing teleportation [7].

Buhrman, Cleve, Wigderson [8] and Cleve, van Dam, Nielsen, and Tapp [9] considered the case of the two-party function computation with quantum communication and used

check for updates

**Citation:** Meng, R.; Ramamoorthy, A. Communication Complexity of Entanglement-Assisted Multi-Party Computation. *Entropy* **2024**, *26*, 896. https://doi.org/10.3390/e26110896

Academic Editor: Osamu Hirota

Received: 31 August 2024
Revised: 17 October 2024
Accepted: 19 October 2024
Published: 23 October 2024

reduction techniques to connect problems in QCC to other known problems, and derived upper/lower bounds for QCC in this manner. In particular, the first work [8] showed examples, such as the set disjointness function, where quantum protocols are strictly better than classical ones in the bounded-error setting. Here, the set-disjointness problem is such that each player has a set and wants to decide if their intersection is empty. Buhrman and de Wolf [10] generalized the two-party "log-rank" lower bound of classical communication complexity to QCC where quantum protocols use both shared entanglement and quantum communication. For other two-party upper/lower bound techniques, see [11–15].

**Related Work:** We will now discuss works in multi-party quantum communication complexity. There are mainly two kinds of models. The number-in-hand (NIH) model assumes each player observes only one variable. The number-on-forehead (NOF) model assumes each player observes all but one variable. François and Shogo [16] considered the NIH model with quantum communication and provided a quantum protocol for a three-party triangle-finding problem; the formulation considers bounded error. This has a polynomial advantage with respect to any classical protocol. Here, the triangle-finding problem is such that the edge set of a graph is distributed over each user and the task is to find a triangle of the graph.

The results in the following two works apply to both NIH and NOF models. Lee, Schechtman, and Shraibman [17] proved a Grothendieck-type inequality and subsequently derived a general lower bound for the multi-party QCC for the Boolean function in Yao's model. Following this work, Briet, Buhrman, Lee, and Vidick [18] showed a similar inequality for the multi-party XOR game and established that the discrepancy method provides lower bounds for QCC when the combined model is of the third type discussed earlier.

Buhrman, van Dam, Høyer, and Tapp [19] considered the NIH model with shared entanglement and proposed a three-party problem with a quantum protocol that is better than any classical protocol by a constant factor. Following this work, Xue, Li, Zhang, Guo [20], and Galv ao [21] showed similar results under the same function with more restrictions. The work most closely related to ours is by Cleve and Buhrman [5]. This study involved three players (Alice, Bob, and Carol) who each have $m$-bit strings, denoted as $\vec{x}, \vec{y}$, and $\vec{z}$, respectively. The strings are such that $\vec{x} + \vec{y} + \vec{z} = \mathbb{1}$, meaning their binary sum (modulo-2) results in the all-ones vector. The goal is for Alice to compute the following:

$$g(x, y, z) = \sum_{i=1}^{m} x_i y_i z_i$$

using binary arithmetic. We note that the communication from Bob and Carol to Alice is purely classical; however, they can use entanglement in a judicious manner. For this particular function, ref. [5] shows that a classical protocol (without entanglement) requires three bits of communication, whereas if the parties share $3m$ entangled qubits, then two bits of communication are sufficient.

**Main Contributions:** In this work, we consider a generalization of the original work of [5]. In particular, we consider a scenario with $n$ players (for prime $n$) that observe values that lie in a higher-order finite field, with a more general promise that is satisfied by the observed values. As we consider more players and higher-order finite fields, the techniques used in the original work are not directly applicable in our setting. For instance, when $n = 3$, our generalized inner product function is defined over $\mathbb{F}_3$ arithmetic (modulo-3), whereas in the same setting, ref. [5] considers binary (modulo-2) arithmetic. Thus, even though we consider a similar problem, we highlight that the result of [5] cannot be recovered as a special case of our result.

Our work provides the following contributions:

- We demonstrate a quantum protocol that allows for the function to be computed with $(n-1)\log n$ bits. We use the quantum Fourier transform as a key ingredient in our method.

- On the other hand, we demonstrate a classical protocol that requires the communication of $(n-1)^2(\log n^2)$ bits.
- To obtain a lower bound on the classical communication complexity, we define an appropriate integer linear programming problem that demonstrates that our quantum protocol is strictly better than any classical protocol.

This paper is organized as follows. Section 2 discusses the problem formulation and Section 3 discusses our quantum protocol. Sections 4 and 5 discuss our classical protocol and the lower bound on any classical protocol, respectively.

## 2. Problem Formulation

### *2.1. Classical/Quantum Communication Scenarios*

Let $\mathcal{X}_i, i = 1, \ldots, n$ and $\mathcal{Y}$ denote the sets in which the inputs and the output lie, and let $f(x_1, \ldots, x_n) : \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \mapsto \mathcal{Y}$ be a multivariate function. There are $n$ players, and the $i$-th player is given $x_i \in \mathcal{X}_i$. The first player (henceforth, Alice) receives information from each of the players; this communication should allow her to compute $f(x_1, \ldots, x_n)$. The players are not allowed to communicate with each other.

In the classical protocol, players 2 to $n$ communicate to Alice via classical channels. In the quantum protocol, we assume that the users have access to shared entanglement as a free resource; however, the communication remains classical. In both scenarios, the classical/quantum communication complexity is defined as the minimum number of classical bits transmitted such that Alice can compute the function among all classical/quantum protocols.

### *2.2. Generalized Inner Product Function with a Promise*

In this work, we consider a specific multivariate function and the setting where $n \geq 3$ (number of players) is prime. Let $\mathbb{F}_n$ denote the finite field of order-$n$ and $[m] \triangleq \{1, \ldots, m\}$. The $i$-th player is given a vector $\vec{x}^i = [x_1^i \ldots x_m^i]^T \in \mathbb{F}_n^m$, i.e., each $x_j^i \in \mathbb{F}_n$. The vectors satisfy the following "promise": $\forall j \in [m]$, the $j$-th component of each player's vector is such that we have the following:

$$[x_j^1, \ldots, x_j^n]^T \in \{a[1, \ldots, 1]^T + b[0, \ldots, n-1]^T | a, b \in \mathbb{F}_n\},$$

i.e., $[x_j^1, \ldots, x_j^n]^T$ lies in a two-dimensional vector space spanned by the basis vectors $[1, \ldots, 1]^T$ and $[0, 1, \ldots, n-1]^T$. In this case, it can be observed that $[x_j^1, \ldots, x_j^n]^T$ is either a multiple of the all-ones vector (if $b = 0$) or a permutation of $[0, 1, \ldots, n-1]$ (if $b \neq 0$). The function to be computed is the generalized inner product function given by the following:

$$GIP(\vec{x}^1, \ldots, \vec{x}^m) = \sum_{i=1}^{m} \left( \prod_{j=1}^{n} x_i^j \right), \tag{1}$$

where the operations are over $\mathbb{F}_n$.

**Remark 1.** *In [5], the promise equivalently means that the j-th component of Alice, Bob, and Carol's vector lies in a two-dimensional affine subspace over $\mathbb{F}_2$ given by $\{a \cdot \vec{v_1} + b \cdot \vec{v_2} \in \mathbb{F}_2^3 : a, b \in \mathbb{F}_2\} + \mathbb{1}$. Thus, the promise introduced in our work can be considered as a natural higher-order generalization of their promise.*

## 3. Proposed Quantum Protocol

We first discuss the entangled states and unitary transforms that will be used in the proposed quantum protocol in Section 3.1. In Section 3.2, we discuss the quantum protocol in detail, including a proof of its correctness. A word about notations: In the following, for complex vectors $\vec{u}, \vec{v}$, $\langle \vec{u}, \vec{v} \rangle = \sum_i u_i^\dagger v_i$ denotes the usual inner product. On the other hand, if $\vec{u}, \vec{v} \in \mathbb{F}_n^m$, then $\langle \vec{u}, \vec{v} \rangle = \sum_{i=1}^m u_i v_i$ denotes the inner product over $\mathbb{F}_n$. Moreover, $\delta_{ij}$ denotes

the Kronecker delta function, which equals 1 if $i = j$ and 0 otherwise. All logarithms in this paper are to base-2.

### *3.1. Entanglement Resource and Unitary Transforms Used*

#### 3.1.1. Shared Entangled States

Consider $n$ isomorphic $n$-dimensional quantum systems, where each system has a computational basis denoted $\mathcal{B} = \{|0\rangle, |1\rangle, \ldots, |n-1\rangle\}$. There are $m$ entangled states shared among $n$ players. For $i \in [m]$, we prepare the entangled state as follows:

$$|\Phi_i\rangle := \frac{1}{\sqrt{n}} \sum_{k=0}^{n-1} |k \ldots k\rangle . \tag{2}$$

The $j$-th subsystem of this entangled state is given to $j$-th player for $j = 1, \ldots, n$.

#### 3.1.2. Quantum Fourier Transform

Let $\omega := e^{\frac{2\pi \mathrm{i}}{n}}$ denote the $n$-th root of unity. The quantum Fourier transform (QFT) is defined by the following unitary map:

$$|j\rangle \mapsto \frac{1}{\sqrt{n}} \sum_{k=0}^{n-1} \omega^{jk} |k\rangle , \forall |j\rangle \in \mathcal{B}. \tag{3}$$

Let $QFT^{\otimes l}$ denote the QFT performed over $l$ isomorphic systems.

#### 3.1.3. Phase Shift Map

For $j \in \mathbb{F}_n$, we define the following:

$$P_0 \triangleq \begin{cases} |0\rangle \mapsto \omega^{-\frac{n-1}{2}} |0\rangle \\ |i\rangle \mapsto |i\rangle , i \neq 0. \end{cases}$$
$$\text{If } j \neq 0, P_j \triangleq |i\rangle \mapsto \omega^{-\frac{1}{n}(ij \bmod n)} |i\rangle . \tag{4}$$

### *3.2. The Quantum Protocol*

Next, we introduce the quantum protocol that uses $(n-1)(\log n)$ bits.

**Theorem 1.** *There exists a quantum protocol for computing $GIP(\vec{x}^1, \ldots, \vec{x}^n)$, which uses $(n-1)\log n$ bits.*

In our protocol (see Algorithm 1), for each $i = 1, \ldots, m$, each player $p$ examines $x_i^p$ and applies the corresponding phase shift map to her subsystem of $|\Phi_i\rangle$. Following this, she applies the QFT on each of her symbols and then measures in the computational basis; this yields $s_i^p \in \mathbb{F}_n$ for $i = 1, \ldots, m$. Player $p$ then transmits $\sum_{i=1}^m s_i^p$. As the players $2 \leq p \leq n$ transmit a symbol from $\mathbb{F}_n$, it is clear that the total communication in the protocol is $(n-1)(\log n)$.

**Algorithm 1:** Proposed quantum protocol.

For $i \in \{1, \ldots, m\}$, prepare maximally entangled "shared state" $|\Phi_i\rangle$ (cf. (2)) and distribute corresponding subsystems to all players.
**for** player $p \in \{1, \ldots n\}$ **do**
  **for** each $i \in \{1, \ldots, m\}$ **do**
    Assume $x_i^p = j$, then player $p$ applies $P_j$ (cf. (4)) on her part of $|\Phi_i\rangle$.
    Player $p$ performs $QFT$ on her part of the shared state.
    Player $p$ measures her part of the shared state in the computational basis, yielding $s_i^p \in \mathbb{F}_n$
  **end for**
  $s^p \leftarrow \sum_{i=1}^m s_i^p$
  Player $p$ sends $s^p$ to Alice if $p \neq 1$
**end for**
$GIP(\vec{x}^1, \ldots, \vec{x}^n) = \sum_p s^p$.

To show the proof of correctness of the protocol, we need the following auxiliary lemma. The proof appears in Appendix A:

**Lemma 1.** *Let $\vec{\alpha} = [1, \ldots, 1]^T \in \mathbb{F}_n^n$. Then, for each $x \in \mathbb{F}_n$, we have the following:*

$$QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-jx}\,|j\cdot\vec{\alpha}\rangle\right) = \frac{1}{n^{\frac{n}{2}}}\sum_{\vec{k}\in\{0,\ldots,n-1\}^n} a_{\vec{k}}\,|\vec{k}\rangle\,. \tag{5}$$

*Then the amplitude $a_{\vec{k}} \neq 0$ iff $\sum_{j=1}^n k_j = x$ where $\vec{k} = [k_1, \ldots, k_n]^T$.*

The proof of correctness of the protocol hinges on the following lemma:

**Lemma 2.**

$$\sum_{p=1}^{n} s_i^p = \prod_{p=1}^{n} x_i^p, \text{for } i = 1, \ldots, m. \tag{6}$$

**Proof.** The state jointly measured by each player is as follows:

$$QFT^{\otimes n}\left(\sum_{j=0}^{n-1}\left(\otimes_{p=1}^n P_{x_i^p}\right)\frac{1}{\sqrt{n}}\,|j\cdot\vec{\alpha}\rangle\right),$$

where $\vec{\alpha} = [1, \ldots, 1]^T$. If $[x_i^1, \ldots, x_i^n]^T = [j, \ldots, j]^T$, then we have the following (see Appendix B for derivation):

$$P_j^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}|k\cdot\vec{\alpha}\rangle\right) \mapsto \frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-kj}\,|k\cdot\vec{\alpha}\rangle\,. \tag{7}$$

Thus, $QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-kj}\,|k\cdot\vec{\alpha}\rangle\right)$ has non-zero coefficients only for states $|\vec{k}\rangle$ such that $\sum_{l=1}^n k_l = j$ by Lemma 1. Therefore, the measurement result $[s_i^1, \ldots, s_i^n]^T$ must be one of $\vec{k} = [k_1, \ldots, k_n]^T$ s.t.

$$\sum_{l=1}^{n} k_l = j \overset{(a)}{=} j^n = \prod_{p=1}^{n} x_i^p$$

where $(a)$ follows from the fact that $j \in \mathbb{F}_n$.

Now, we assume the following: $[x_j^1, \ldots, x_j^n]^T = a[1, \ldots, 1]^T + b[0, 1, \ldots, n-1]^T$ with $b \neq 0$. This implies that $a + b \cdot i \in \mathbb{F}_n$ is distinct for each $i \in \{0, \ldots, n\}$. It can be observed that $a[1, \ldots, 1] + b[0, \ldots, n-1]$ is a permutation of $[0, \ldots, n-1]$, so it suffices to discuss

$[x_i^1, \ldots, x_i^p]^T = [0, \ldots, n-1]^T$ by symmetry. We have the following (see Appendix B for derivation):

$$P_0 \otimes \cdots \otimes P_{n-1}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}|k\cdot\vec{\alpha}\rangle\right) \mapsto \frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\sum_{k=0}^{n-1}|k\cdot\vec{\alpha}\rangle \tag{8}$$

We have the following:

$$QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\sum_{k=0}^{n-1}|k\cdot\vec{\alpha}\rangle\right) = \omega^{-\frac{n-1}{2}}QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}|k\cdot\vec{\alpha}\rangle\right) = \frac{1}{n^{\frac{n}{2}}}\omega^{-\frac{n-1}{2}}\sum_{\vec{k}\in\mathbb{F}_n^n} a_{\vec{k}}\,|\vec{k}\rangle\,.$$

By Lemma 1, $a_{\vec{k}} \neq 0$ iff $\sum_{l=1}^n k_l = 0$. Therefore, the measurement result $[s_i^1, \ldots, s_i^n]^T$ must be $\vec{k} = [k_1, \ldots, k_n]^T$ such that

$$\sum_{l=1}^{n} k_l = 0 = \prod_{j=0}^{n-1} j = \prod_{p=1}^{n} x_i^p.$$

□

Now, we show that our protocol computes $GIP(\vec{x}^1, \ldots, \vec{x}^n)$ correctly. Since $s^p = \sum_i s_i^p$, by applying (6), we have the following:

$$\sum_p s^p = \sum_p \sum_i s_i^p = \sum_i \sum_p s_i^p = \sum_i \prod_{p=1}^{n} x_i^p = GIP(\vec{x}^1, \ldots, \vec{x}^n).$$

## 4. Proposed Classical Protocol

We now consider purely classical protocols for our problem, i.e., ones that do not consider entanglement. At the top level, our classical scheme operates by communicating the "number" of different symbols that exist within each player's vector. We show that this suffices for Alice to recover the function value.

More precisely, let $\beta_k^p$ be the number of "$k$" values in the vector of $p$-th player; recall that player $p$ is assigned $\vec{x}_p = x_1^p \ldots x_m^p$. Note that $\sum_{k=0}^{n-1} \beta_k^p = m$.

**Theorem 2.** *There exists a classical protocol for computing $GIP(\vec{x}^1, \ldots, \vec{x}^n)$ that uses $(n-1)^2(\log n^2)$ bits.*

In our protocol (see Algorithm 2), for each $i = 1, \ldots, n-1$, each player $p$ transmits $\beta_k^p \pmod{n^2}$. Alice computes each $\beta_0^p \pmod{n^2}$ by using the fact that $\sum_{k=0}^{n-1} \beta_k^p = m$. Finally, Alice computes the value of the function by using $\{\beta_k^p \pmod{n^2} | k \in \mathbb{F}_n, p \in [n]\}$ For each player $p \in \{2, \ldots, n\}$, $p$ transmits $\{\beta_1^p \pmod{n^2}, \ldots, \beta_{n-1}^p \pmod{n^2}\}$. The total number of bits transmitted is $(n-1)^2(\log n^2)$. The proof of the Theorem 2 appears in Appendix C.

**Algorithm 2:** Proposed Classical protocol.

**for** player $p \in \{1, \dots n\}$ **do**
  **for** each $k \in \mathbb{F}_n$ **do**
    $\beta_k^p \leftarrow$ number of "$k$" values in $x_1^p \dots x_m^p$
    **if** $p$ is not Alice **and** $k \neq 0$ **then**
      $p$ sends $\beta_k^p \pmod{n^2}$ to Alice
    **end if**
  **end for**
**end for**
**for** $p \in \{2, \dots n\}$ **do**
  Alice computes $\beta_0^p \pmod{n^2}$ by using
  $\sum_{k=0}^{n-1} \beta_k^p = m$.
**end for**
$W \leftarrow \sum_{p=1}^{n} \sum_{k=1}^{n-1} k \cdot \beta_k^p + \frac{n^2-n}{2}(n-1) \sum_{p=1}^{n} \beta_0^p \pmod{n^2}$
$GIP(\vec{x}^1, \dots, \vec{x}^n) = W/n$

## 5. Classical Communication Complexity Lower Bound

We now discuss a lower bound on the communication complexity of *any* classical protocol that demonstrates a strict separation between our proposed quantum protocol and any classical protocol. Analytically, this seems to be a rather hard problem, and we discuss it as an item for future work. We can show, however, the strict separation numerically using ILPs (see Section 5.1 below). In addition, we present an analytical argument below that demonstrates that for $n = 3$, the communication complexity of any classical protocol is at least $2 \log_2 3$.

We assume that Alice, Bob, and Carol are given vectors $\vec{x}^1$, $\vec{x}^2$, and $\vec{x}^3$, respectively, each of length $m$. The promise (cf. Section 2.2) is equivalent to the following:

$$x_j^1 + x_j^2 + x_j^3 = 0, \text{for } j = 1, \dots, m. \tag{9}$$

This implies that the GIP function in this case can be computed if we know any two out of $\vec{x}^1$, $\vec{x}^2$, and $\vec{x}^3$. We assume that Carol labels her sequences ($\vec{x}^3 \in \{0,1,2\}^m$) with one of—at most—three possible labels. We denote this label by a mapping $\beta : \{0,1,2\}^m \to \{0,1,2\}$. Recall that Alice knows her sequence $\vec{x}^1$.

**Definition 1.** *We define Bob's confusion graph $G_B = (V_B, E_B)$ as follows. The vertex set $V_B$ corresponds to the $3^m$ sequences $\vec{x}^2 \in \{0,1,2\}^m$. The $i$-th such sequence is denoted by $\vec{x}^2[i]$ for $i = 0, \dots, 3^m - 1$, with similar notations applied to the sequences for Alice and Carol.*

*There exists an edge $(\vec{x}^2[i], \vec{x}^2[j]) \in E_B$, for $i \neq j$ if there exists an Alice sequence $\vec{x}^1[*]$ and Carol sequences $\vec{x}^3[a]$ and $\vec{x}^3[b]$, such that (i) $\beta(\vec{x}^3[a]) = \beta(\vec{x}^3[b])$ (note that we allow $a = b$), and (ii) $GIP(\vec{x}^1[*], \vec{x}^2[i], \vec{x}^3[a]) \neq GIP(\vec{x}^1[*], \vec{x}^2[j], \vec{x}^3[b])$.*

Note that if $(\vec{x}^2[i], \vec{x}^2[j]) \in E_B$, Bob must assign different labels to $\vec{x}^2[i]$ and $\vec{x}^2[j]$; otherwise, Alice has no way to compute the function with zero error. The concept of the confusion graph dates back to the work of Shannon [22].

The main concept of the argument below is to show that there exists a triangle in $G_B$. This implies that Bob needs to use at least three labels for Alice to decode with zero error.

Since Carol uses, at most, three labels, the pigeon-hole principle dictates that there must be at least $3^{m-1}$ sequences that share the same Carol label. Let us denote this set by $\mathcal{C}$.

**Claim 1.** *There is a subset of two coordinates where all nine patterns $\{0,1,2\}^2$ appear within the sequences in $\mathcal{C}$.*

**Proof.** Suppose that $m$ is even. Then, we can partition the coordinates as $\{1,2\},\{3,4\},\ldots,\{m-1,m\}$. Let us arrange the sequences in $\mathcal{C}$ as rows; the number of rows is $|\mathcal{C}| \geq 3^{m-1}$. Now, suppose that the projection onto any pair of coordinates has, at most, 8 representatives, then, the size of $\mathcal{C}$ can be, at most, $8^{m/2}$. Now, we have the following:

$$\begin{aligned}\frac{3^{m-1}}{8^{m/2}} &= \frac{1}{3} \times \left(\frac{9}{8}\right)^{m/2} \\ &> 1,\end{aligned}$$

for large enough $m$. □

Without loss of generality, we assume that all nine patterns occur within the first two coordinates of $\mathcal{C}$. We pick nine of such representatives from $\mathcal{C}$ and denote them as $\mathbf{z}_{00}, \mathbf{z}_{01}, \mathbf{z}_{02}, \mathbf{z}_{10}, \ldots, \mathbf{z}_{22}$; the subscripts correspond to the values on the first two coordinates.

Let us pick Alice's sequence $\vec{x}^1[*] = [110\ldots0]$. Corresponding to this $\vec{x}^1[*]$, for the Carol sequences $\mathbf{z}_{00}, \ldots, \mathbf{z}_{22}$, using the given promise, we can determine the corresponding Bob sequences $\mathbf{y}_{00}, \ldots, \mathbf{y}_{22}$. We note the following:

$$\begin{aligned}\mathbf{y}_{00} &= [22 - z_{00}(3:m)], \\ \mathbf{y}_{01} &= [21 - z_{01}(3:m)], \text{ and} \\ \mathbf{y}_{11} &= [11 - z_{11}(3:m)].\end{aligned}$$

where $z_{00}(3:m)$ denotes the components of vector $z_{00}$ from index 3 onward (basically the MATLAB notation).

**Claim 2.** *In Bob's confusion graph, $G_B$, the sequences $\mathbf{y}_{00}, \mathbf{y}_{01}$, and $\mathbf{y}_{11}$ form a triangle.*

**Proof.** We need to examine $GIP(\vec{x}^1[*], \mathbf{y}_i, \mathbf{z}_i)$ for $i = 00, 01, 11$. Since only the first two coordinates matter, given $\vec{x}^1[*] = [110\ldots0]$, the corresponding evaluations are $0, 1, 2$, which are pairwise different. □

This argument shows that Bob must use at least three labels for Alice to decode with zero error. By symmetry, Carol must also use three labels. To summarize, the communication complexity of any classical protocol is at least $2\log 3$ bits.

**Remark 2.** *It may be possible to use a variant of the above combinatorial argument to establish that the chromatic number of $G_B$ is strictly larger than three. However, this does not seem to follow in a straightforward manner.*

### *5.1. ILP Feasibility Problem for Classical Lower Bound*

We now present a lower bound on the communication complexity of any deterministic classical protocol for our problem. To this end, we frame this as an integer linear programming problem (ILP) that can be solved numerically. The primary aim of the ILP is to establish a correspondence between each deterministic classical protocol and a feasible point within the ILP. Therefore, the feasibility of the ILP, which can be numerically verified, implies the existence of a deterministic classical protocol and vice versa (this correspondence between the ILP and classical protocols is valid only for the deterministic case. The ILP does not account for randomized protocols where players may have access to public and/or private randomness).

Suppose, for $p \in \{2,\ldots,n\}$, the $p$-th player sends symbols (labels) in $[l^p] := \{1,2,\ldots,l^p\}$ for some large enough positive integer $l^p$. Let $c \in [l^p]$ and define $I_{\vec{x}^p,c} \in \{0,1\}$ to be the indicator that the $p$-th player sends $c$ when it has the vector $\vec{x}^p \in \mathbb{F}_n^m$. As this mapping is unique, we have $\sum_{c\in[l^p]} I_{\vec{x}^p,c} = 1$. Furthermore, for a given set of vectors $\vec{x}^p$ for $p \in \{2,\ldots,n\}$, if the $p$-th player sends label $c^p$, we have $\prod_{p=2}^{n} I_{\vec{x}^p,c^p} = 1$.

Consider two sets of vectors $\{\vec{x}^p \in \mathbb{F}_n^m | p \in \{1,\ldots,n\}\}$, $\{\vec{z}^p \in \mathbb{F}_n^m | p \in \{1,\ldots,n\}\}$. We denote the following:

$$(\vec{x}^1,\ldots,\vec{x}^n) \sim_{GIP} (\vec{z}^1,\ldots,\vec{z}^n)$$

if the following conditions are satisfied.

1. Both $(\vec{x}^1,\ldots,\vec{x}^n)$ and $(\vec{z}^1,\ldots,\vec{z}^n)$ satisfy the promise (cf. Section 2.2).
2. $\vec{x}^1 = \vec{z}^1$.
3. $GIP(\vec{x}^1,\ldots,\vec{x}^n) \neq GIP(\vec{z}^1,\ldots,\vec{z}^n)$.

This definition applies to distinct inputs with the "same" Alice vector, but different function evaluations. It can be seen that—for two such distinct inputs—the symbols communicated by players 2 to $n$ have to be distinct, otherwise, Alice has no way to decode in a zero-error fashion.

Our proposed ILP works with fixed $l^p$'s and a fixed value of $m$. Due to complexity reasons, $m$ cannot be very large. However, if the ILP is infeasible for a given $l^p$ and a $\tilde{m}$, then our lower bound holds for arbitrary values $m \geq \tilde{m}$. Our lower bound would continue to hold even if Alice was provided the values $x^p_{\tilde{m}+1},\ldots,x^p_m$ for all players $p = 2,\ldots,n$.

Consider the following $0-1$ integer programming feasibility problemL

$$\begin{aligned} & \min \quad 0 \\ \text{s.t.} \quad & p \in \{2,\ldots,n\}, c \in [l^p], \vec{x}^p \in \mathbb{F}_n^m, \\ & I_{\vec{x}^p,c} \in \{0,1\}, \\ & \sum_{c\in[l^p]} I_{\vec{x}^p,c} = 1, \forall \vec{x}^p, \\ & \sum_{c^2\in[l^2],\ldots,c^n\in[l^n]} |\prod_{p=2}^{n} I_{\vec{x}^p,c^p} - \prod_{p=2}^{n} I_{\vec{z}^p,c^p}| = 2 \\ & \text{for all } (\vec{x}^1,\ldots,\vec{x}^n) \sim_{GIP} (\vec{z}^1,\ldots,\vec{z}^n). \end{aligned} \tag{10}$$

The infeasibility of the above integer programming problem corresponds to a lower bound on the classical communication complexity. The proof of the following theorem appears in Appendix D.

**Theorem 3.** *There exists a deterministic classical protocol computing $GIP(\cdot)$ where each player sends—at most—$l^p$ different labels for $p \in \{2,\ldots,n\}$ iff the above integer programming is feasible.*

**Remark 3.** *The above integer program contains constraints that involve the product of variables and equality constraints with sums of absolute values. We show how these constraints can be linearized in Appendix E. The entire code for our ILP is available in this online repository [23].*

### *5.2. Numerical Experiments*

In our numerical experiments, we considered an instance of the ILP involving $n = 3$ players, namely Alice, Bob, and Carol. Let $m$ represent the length of each vector, while $[l^b]$, $[l^c]$ denote the sets of labels used by Bob and Carol, with $l^b, l^c$ denoting the sizes of these sets.

We assume that Alice, Bob, and Carol are given vectors $\vec{x}^1$, $\vec{x}^2$, and $\vec{x}^3$, respectively, each of length $m$. In this case, the promise is given by the following: (9). It can be observed that swapping the vectors of Bob and Carol still satisfies the promise. Due to this inherent symmetry, a protocol with communication lengths $l^b = x$ and $l^c = y$ exhibits the same feasibility as one with $l^b = y$ and $l^c = x$. Consequently, for the ILP we can assume that $l^b \leq l^c$.

The experimental results under varying settings of $l^b, l^c, m$ are displayed in Table 1. For instance, it shows that when $l^b = 1$ and $l^c = 17$, the ILP is infeasible with $m = 3$. This implies that for a feasible classical protocol, with $l^b = 1$, we need at least $\log(18)$ bits to be

transmitted from Carol. Similarly, the triplets $(m, l^b, l^c) = (2, 2, 4)$ and $(3, 3, 3)$ are infeasible. This implies that when $l^b$ equals 2 or 3, the sum rate is $\geq \min(\log 2 + \log 5, \log 3 + \log 4)$.

**Table 1.** Numerical results.

| $m$ | $l^b$ | $l^c$ | **Feasibility** |
|---|---|---|---|
| 1 | 1 | 3 | Feasible |
| 3 | 1 | 17 | Infeasible |
| 2 | 2 | 4 | Infeasible |
| 3 | 3 | 3 | Infeasible |
| 2 | 3 | 4 | Feasible |
| 3 | 5 | 5 | Feasible |

Recalling that our proposed protocol employs $2\log(3)$ bits of communication, and by the fact that

$$2\log 3 < \min(\log 18, \log 3 + \log 4, \log 2 + \log 5)$$

we conclude that there is a strict separation between our quantum protocol and any classical protocol. We note here that we have expressed the communication complexity of both protocols in terms of bits by converting to base-2 logarithms. However, it is important to interpret the results, e.g., the quantum protocol is feasible if Bob and Carol use ternary communication (one of three possible symbols). Conversely, the classical protocol requires that at least one of Bob or Carol transmit one of four possible symbols. In this sense, the quantum protocol is strictly better.

## 6. Conclusions

We considered the communication complexity problem of the GIP function under a specific promise. We proposed a quantum protocol utilizing $(n-1)\log(n)$ bits and a classical protocol employing $(n-1)^2(\log n^2)$ bits. By establishing a connection between the integer linear programming feasibility problem and the existence of a classical protocol with a particular communication complexity, we were able to provide numerical evidence supporting the quantum advantage in our model's communication complexity. The main limitation of our work is the absence of an analytical lower bound on the classical communication complexity of our problem.

It would be interesting to analytically investigate the quantum advantage in the asymptotic limit as $n$ increases, and to consider promises where the quantum advantage can be analytically demonstrated, or other problems that also encompass the work in [5] as a special case.

**Author Contributions:** Conceptualization, R.M. and A.R.; Methodology, R.M. and A.R.; Writing—review & editing, R.M. and A.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** The material in this work was supported in part by NSF grants CIF-1910840 and CIF-2115200.

**Institutional Review Board Statement:** Not applicable.

**Data Availability Statement:** There is no significant data associated with the study. The corresponding author can be contacted if needed.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A. Proof of Lemma 1

Recall that

$$\vec{\alpha} = [1, \ldots, 1]^T.$$

The action of $\mathrm{QFT}^{\otimes n}$ on $\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj}\left|j\cdot\vec{\alpha}\right\rangle$ is

$$\frac{1}{n^{\frac{n}{2}}}\sum_{\vec{k}\in\mathbb{F}_n^n}a_{\vec{k}}\left|\vec{k}\right\rangle=QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj}\left|j\cdot\vec{\alpha}\right\rangle\right)=\frac{1}{n^{\frac{n}{2}}}\sum_{\vec{k}\in\mathbb{F}_n^n}\left(\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj}\cdot\omega^{\langle\vec{k},j\cdot\vec{\alpha}\rangle}\right)\left|\vec{k}\right\rangle.$$

Write $\vec{k}=[k_1,\ldots,k_n]^T$. Therefore,

$$a_{\vec{k}}=\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj}\cdot\omega^{\langle\vec{k},j\cdot\vec{\alpha}\rangle}=\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj+j\sum_{l=1}^{n}k_l}.$$

When $\vec{k}$ satisfies $\sum_{l=1}^{n}k_l=x$, we have that $a_{\vec{k}}=\sqrt{n}\neq 0$. Otherwise, since $-x+\sum_{l=1}^{n}k_l\neq 0$ and $n$ are prime, $1-\omega^{n(-x+\sum_{l=1}^{n}k_l)}=0$ and $1-\omega^{-x+\sum_{l=1}^{n}k_l}\neq 0$. We have the following:

$$\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{-xj+j\sum_{l=1}^{n}k_l}=\frac{1}{\sqrt{n}}\sum_{j=0}^{n-1}\omega^{(-x+\sum_{l=1}^{n}k_l)j}=\frac{1}{\sqrt{n}}\frac{1-\omega^{n(-x+\sum_{l=1}^{n}k_l)}}{1-\omega^{-x+\sum_{l=1}^{n}k_l}}=0.$$

## Appendix B. Derivation of Equations (7) and (8)

We derive Equation (7) by considering two cases. The first case occurs when $[x_i^1,\ldots,x_i^n]^T=[0,\ldots,0]^T$. Then, we have the following:

$$P_0^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle\right)\mapsto\frac{1}{\sqrt{n}}\omega^{-\frac{n(n-1)}{2}}\left|0\cdot\vec{\alpha}\right\rangle+\frac{1}{\sqrt{n}}\sum_{k=1}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle=\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle=\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-k\cdot 0}\left|k\cdot\vec{\alpha}\right\rangle. \quad \text{(A1)}$$

The second case occurs when $[x_i^1,\ldots,x_i^n]^T=[j,\ldots,j]^T$ with $j\neq 0$. Then, we have the following:

$$P_j^{\otimes n}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle\right)\mapsto\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-n(\frac{1}{n}(kj \bmod n))}\left|k\cdot\vec{\alpha}\right\rangle=\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-kj}\left|k\cdot\vec{\alpha}\right\rangle \quad \text{(A2)}$$

where the last equality holds since $\omega^{-kj}=\omega^{-(kj \bmod n)}$. Thus, in this case, collectively, we can express the joint state after phase-shifting as $\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\omega^{-kj}\left|k\cdot\vec{\alpha}\right\rangle$.

Now, we derive Equation (8). We have the following:

$$\begin{aligned}
&P_0\otimes\cdots\otimes P_{n-1}\left(\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle\right)\mapsto\\
&\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\left|0\cdot\vec{\alpha}\right\rangle+\frac{1}{\sqrt{n}}\sum_{k=1}^{n-1}\omega^{-\sum_{j=0}^{n-1}\frac{1}{n}(kj \bmod n)}\left|k\cdot\vec{\alpha}\right\rangle\\
&\overset{(a)}{=}\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\left|0\cdot\vec{\alpha}\right\rangle+\frac{1}{\sqrt{n}}\sum_{k=1}^{n-1}\omega^{-\frac{1}{n}(\sum_{j=0}^{n-1}j)}\left|k\cdot\vec{\alpha}\right\rangle\\
&=\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\left|0\cdot\vec{\alpha}\right\rangle+\frac{1}{\sqrt{n}}\sum_{k=1}^{n-1}\omega^{-\frac{1}{n}[\frac{n(n-1)}{2}]}\left|k\cdot\vec{\alpha}\right\rangle\\
&=\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle
\end{aligned}$$

where $(a)$ follows from the fact that $\{kj \bmod n | j\in\{0,\ldots,n-1\}\}=\{0,\ldots,n-1\}$ for $k\neq 0$ (we emphasize that this statement only holds if $n$ is prime) . We have the following;

$$QFT^{\otimes n}\left(\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle\right)=\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}QFT^{\otimes n}\left(\sum_{k=0}^{n-1}\left|k\cdot\vec{\alpha}\right\rangle\right)=\frac{1}{\sqrt{n}}\omega^{-\frac{n-1}{2}}\sum_{\vec{k}\in\mathbb{F}_n^n}a_{\vec{k}}\left|\vec{k}\right\rangle.$$

## Appendix C. Proof of Theorem 2

For $a, b \in \mathbb{F}_n$, we define $M_{a,b} = \{i \in [m] | [x_i^1, \ldots, x_i^n]^T = a[1, \ldots, 1]^T + b[0, 1, \ldots, n-1]^T\}$ and $m_{a,b} = |M_{a,b}|$. Since the promise ensures that, for each $i \in \{1, \ldots, m\}$, there exists $a, b \in \mathbb{F}_n$ s.t. $[x_i^1, \ldots, x_i^n]^T = a[1, \ldots, 1]^T + b[0, 1, \ldots, n-1]^T \in \mathbb{F}_n^n$, we have that $\{M_{a,b} | a, b \in \mathbb{F}_n\}$ forms a partition of the set $\{1, \ldots, m\}$.

When $i \in M_{a,0}$, i.e., $[x_i^1, \ldots, x_i^n]^T = a[1, \ldots, 1]^T$, we have the following:

$$\prod_{p=1}^{n} x_i^p = a^n = a. \tag{A3}$$

Otherwise, we have $i \in M_{a,b}$ for some $b \neq 0$, so $[x_i^1, \ldots, x_i^n]^T = a[1, \ldots, 1]^T + b[0, \ldots, n-1]^T$. Then, we have the following:

$$\prod_{p=1}^{n} x_i^p = \prod_{i=0}^{n-1} (a + i \cdot b) = 0. \tag{A4}$$

By (A3) and (A4), if $i \in M_{a,b}$, then $\prod_{p=1}^{n} x_i^p = \delta_{0b} \cdot a$. Define

$$\mathbb{1}((x_i^1, \ldots, x_i^n), M_{a,b}) = \begin{cases} 1, & i \in M_{a,b} \\ 0, & \text{otherwise,} \end{cases}$$

i.e., it is the indicator of $i \in M_{a,b}$. Since $i \in M_{a,b}$ for exactly one choice of $(a, b)$, we have the following:

$$\prod_{p=1}^{n} x_i^p = \delta_{0b} \cdot a = \sum_{a,b=0}^{n-1} \delta_{0b} \cdot a \cdot \mathbb{1}((x_i^1, \ldots, x_i^n), M_{a,b}). \tag{A5}$$

We have the following:

$$\begin{aligned} &\sum_{i=1}^{m} \prod_{p=1}^{n} x_i^p \\ &\overset{(*)}{=} \sum_{i=1}^{m} \sum_{a,b=0}^{n-1} a \cdot \delta_{0b} \cdot \mathbb{1}((x_i^1, \ldots, x_i^n), M_{a,b}) \\ &= \sum_{a,b=0}^{n-1} \sum_{i=1}^{m} a \cdot \delta_{0b} \cdot \mathbb{1}((x_i^1, \ldots, x_i^n), M_{a,b}) \\ &= \sum_{a,b=0}^{n-1} a \cdot \delta_{0b} \cdot m_{a,b} = \sum_{a=0}^{n-1} a \cdot m_{a,0} \pmod{n}. \end{aligned} \tag{A6}$$

Here, (*) follows from (A5). Our next step is to show $\sum_{a=0}^{n-1} a m_{a,0} = W/n \pmod{n}$; $W$ is defined in Algorithm 2.

Suppose $i \in M_{a,b}$, then $(x_i^1, \ldots, x_i^n) = a(1, \ldots, 1) + b(0, \ldots, n-1)$. For the $p-$th player, $a + (p-1)b$ is the value of the $i-$th coordinate of the vector $\vec{x}^p$. For a fixed $k \in \mathbb{F}_n$, the set of $(a, b)$ s.t. $a + (p-1)b = k$ is $\{(k, 0), (k + p - 1, -1), (k + 2p - 2, -2), \ldots, (k + (n-1)(p-1), -(n-1)\}$. We have the following: $\forall p \in \{1, \ldots, n\}, k \in \mathbb{F}_n$,

$$\beta_k^p = \sum_{i=0}^{n-1} m_{k+i(p-1),-i}. \tag{A7}$$

Consider $\sum_{i=0}^{n-1}\sum_{p=1}^{n} m_{k+i(p-1),-i}$. When $i=0$, $m_{k+i(p-1),-i}$ is counted $n$ times. Denote $S=\{(0,0),\ldots,(n-1,0)\}$. For arbitrary $[x,y]^T \in \mathbb{F}_n^2 - S$, the equation

$$\begin{bmatrix} k+i(p-1) \\ -i \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$$

has a unique solution given by the following:

$$\begin{aligned} i &= -y, \text{ and} \\ p &= \frac{y-x+k}{y}. \end{aligned}$$

Thus, we have the following:

$$\mathbb{F}_n^2 - S = \{(k+i(p-1), -i | p \in \{1,\ldots,n\}, i \in \{1,\ldots,n-1\}\}.$$

Therefore, when $i \neq 0$, $m_{k+i(p-1),-i}$ is counted exactly once. We have the following:

$$\begin{aligned} &\sum_{p=1}^{n}\sum_{i=0}^{n-1} m_{k+i(p-1),-i} \\ =&\sum_{p=1}^{n}\sum_{i=1}^{n-1} m_{k+i(p-1),-i} + \sum_{p=1}^{n} m_{k+0(p-1),0} \\ =&\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n \cdot m_{k,0}. \end{aligned} \tag{A8}$$

Now, we have the following:

$$\begin{aligned} \sum_{p=1}^{n}\sum_{k=1}^{n-1} k\cdot\beta_k^p &\overset{(A7)}{=} \sum_{k=1}^{n-1} k \sum_{p=1}^{n}\sum_{i=0}^{n-1} m_{k+i\cdot(p-1),-i} \\ &\overset{(A8)}{=} \sum_{k=1}^{n-1} k\Big[\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n\cdot m_{k,0}\Big] \\ &= \frac{n^2-n}{2}\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n\sum_{k=1}^{n-1} k\cdot m_{k,0} \end{aligned}$$

$$\text{and } \sum_{p=1}^{n}\beta_0^p \overset{(A7)}{=} \sum_{p=1}^{n}\sum_{i=0}^{n-1} m_{i\cdot(p-1),-i} \overset{(A8)}{=} \sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n\cdot m_{0,0}.$$

Note $n>2$ is prime, so $n-1$ is divisible by 2. We have the following:

$$\begin{aligned} W =& \sum_{p=1}^{n}\sum_{k=1}^{n-1} k\cdot\beta_k^p + \frac{n^2-n}{2}\cdot(n-1)\sum_{p=1}^{n}\beta_0^p \\ =& \frac{n^2-n}{2}\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n\sum_{k=1}^{n-1} k\cdot m_{k,0} + \frac{n^2-n}{2}\cdot(n-1)\Big[\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n\cdot m_{0,0}\Big] \\ =& n\sum_{k=1}^{n-1} k\cdot m_{k,0} + n^2\cdot\frac{(n-1)}{2}\sum_{a,b\in\mathbb{F}_n^2-S} m_{a,b} + n^2\cdot\frac{(n-1)^2}{2}\cdot m_{0,0} \\ =& n\sum_{k=0}^{n-1} k\cdot m_{k,0} \pmod{n^2}. \end{aligned}$$

When we divide both sides by $n$, we obtain $W/n = \sum_{k=1}^{n-1} k m_{k,0} \pmod{n}$. We are now done because of (A6).

## Appendix D. Proof of Theorem 3

Suppose we have a protocol that computes the function with zero error. Our protocol is deterministic, so for each $\vec{x}^p \in \mathbb{F}_n^m$, it associates exactly one label $\hat{c}$ such that the $p$-th player sends $\hat{c}$ if his vector is $\vec{x}^p$. We set $I_{\vec{x}^p,\hat{c}} = 1$ and $I_{\vec{x}^p,c} = 0$ for all $c \neq \hat{c}$. Therefore, $I_{\vec{x}^p,c} \in \{0,1\}$ and $\sum_{c\in[l]} I_{\vec{x}^p,c} = 1$ are satisfied for all choices of $\vec{x}^p, c$.

Next, suppose we have that $(\vec{x}^1,\ldots,\vec{x}^n) \sim_{GIP} (\vec{z}^1,\ldots,\vec{z}^n)$. Furthermore, assume that the $p$-th player sends $s^p/t^p$ for $\vec{x}^p/\vec{z}^p$ for $p = 2,\ldots,n$. This implies that $\prod_{p=2}^n I_{\vec{x}^p,s^p} = 1$ and that $\prod_{p=2}^n I_{\vec{z}^p,t^p} = 1$. In addition, note that $\exists p$, such that $s^p \neq t^p$ for these sequences, the symbols transmitted from users $2,\ldots,n$ have to be distinct.

Thus, we have the following:

$$1 = |\prod_{p=2}^n I_{\vec{x}^p,s^p} - \prod_{p=2}^n I_{\vec{z}^p,s^p}| = |\prod_{p=2}^n I_{\vec{x}^p,t^p} - \prod_{p=2}^n I_{\vec{z}^p,t^p}|$$

and, consequently, we have the following:

$$\sum_{c_2\in[l^2],\ldots,c_n\in[l^n]} |\prod_{p=2}^n I_{\vec{x}^p,c^p} - \prod_{p=2}^n I_{\vec{z}^p,c^p}| =$$
$$|\prod_{p=2}^n I_{\vec{x}^p,\hat{s}^p} - \prod_{p=2}^n I_{\vec{z}^p,\hat{s}^p}| + |\prod_{p=2}^n I_{\vec{x}^p,\hat{t}^p} - \prod_{p=2}^n I_{\vec{z}^p,\hat{t}^p}| = 2.$$

Therefore, the third constraint is satisfied.

Conversely, if $\{I_{\vec{x}^p,c}|\vec{x}^p \in \mathbb{F}_n^m, c \in [l^p]\}$ satisfies the constraints of the ILP, then we construct a classical protocol as follows. Suppose the $p$-th player has vector $\vec{x^p}$ for $p \in \{1,\ldots,n\}$. Since there exists exactly one $s^p \in [l^p]$ s.t. $I_{x^p,s^p} = 1$, then $p$ sends $s^p$ to Alice for $p \in \{2,\ldots,n\}$. When Alice receives the symbols $s^p, p = 2,\ldots,n$ from the other players, she picks arbitrary $\{\vec{y}^p \in \mathbb{F}_n^m\}_{i=2}^n$ s.t. $(\vec{x}^1,\vec{y}^2,\ldots,\vec{y}^n)$ satisfies the promise and $\forall p \in \{2,\ldots,n\}, I_{\vec{y}^p,s^p} = 1$.Then, she outputs $f(\vec{x}^1,\vec{y}^2,\ldots,\vec{y}^n)$. In what follows, we show that

$$GIP(\vec{x}^1,\vec{y}^2,\ldots,\vec{y}^n) = GIP(\vec{x}^1,\vec{x}^2,\ldots,\vec{x}^n).$$

To see this, assume otherwise. Then, we have the following: $GIP(\vec{x}^1,\vec{y}^2,\ldots,\vec{y}^n) \neq GIP(\vec{x}^1,\vec{x}^2,\ldots,\vec{x}^n)$. We have the following:

$$(\vec{x}^1,\vec{x}^2,\ldots,\vec{x}^n) \sim_{GIP} (\vec{x}^1,\vec{y}^2,\ldots,\vec{y}^n).$$

Owing to the third constraint, we have the following:

$$\sum_{c^2,\ldots,c^n\in[l]} |\prod_{p=2}^n I_{\vec{x}^p,c^p} - \prod_{p=2}^n I_{\vec{y}^p,c^p}| = 2.$$

However, we have that $I_{\vec{x}^i,s^i} = I_{\vec{y}^i,s^i} = 1$ for all $i \in \{2,\ldots,n\}$. By the first and second constraints, we have that $I_{\vec{x}^i,c^i} = I_{\vec{y}^i,c^i} = 0$ for all $i \in \{2,\ldots,n\}$ and $c^i \neq s^i$. Therefore,

$$\sum_{c^2,\ldots,c^n\in[l]} |\prod_{p=2}^n I_{\vec{x}^p,c^p} - \prod_{p=2}^n I_{\vec{y}^p,c^p}| = 0.$$

This gives the desired direction.

## Appendix E. Linearizing Constraints in the Integer Programming Problem

One issue with the optimization problem in (10) is that the third constraint has multiple absolute values and products of variables. Here, we transform the constraints and add extra variables in (10) to obtain the desired ILP.

Our first step is to introduce auxiliary 0–1 variables that correspond to the products of other 0–1 variables. For instance, it can be verified that we can handle $\prod_{i=1}^{k} x_i = x'$ as follows:

$$x' \leq x_i, \text{ for } i = 1, \ldots, k \tag{A9}$$

$$x' \geq \sum_{i=1}^{k} x_i - (k-1). \tag{A10}$$

As a first step, we introduce such auxiliary variables for all terms that involve products of our indicator function in (10).

Following this step, we are left with handling constraints that involve sums of absolute values of differences. For this step, we show how to replace each absolute value difference with another auxiliary variable. In particular, we can replace $|x - y|$ by $z$ as follows:

$$|x - y| = |x - y|^2 = x^2 + y^2 - 2xy = x + y - 2xy$$

where the last step follows from the fact that the variables are of type 0–1. The product term $2xy$ can be linearized as described previously. Following these steps, all constraints in the integer programming problem are linear.

## References


1. Yao, A.C.C. Some Complexity Questions Related to Distributive Computing(Preliminary Report). In Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing, STOC '79, Atlanta, GA, USA, 30 April–2 May 1979; ACM: New York, NY, USA, 1979; pp. 209–213. [CrossRef]
2. Chi-Chih Yao, A. Quantum circuit complexity. In Proceedings of the 1993 IEEE 34th Annual Foundations of Computer Science, Palo Alto, CA, USA, 3–5 November 1993; pp. 352–361. [CrossRef]
3. Horodecki, R.; Horodecki, P.; Horodecki, M.; Horodecki, K. Quantum entanglement. *Rev. Mod. Phys.* **2009**, *81*, 865–942. [CrossRef]
4. Buhrman, H.; Cleve, R.; Massar, S.; de Wolf, R. Nonlocality and communication complexity. *Rev. Mod. Phys.* **2010**, *82*, 665–698. [CrossRef]
5. Cleve, R.; Buhrman, H. Substituting quantum entanglement for communication. *Phys. Rev. A* **1997**, *56*, 1201–1204. [CrossRef]
6. de Wolf, R. Quantum communication and complexity. *Theor. Comput. Sci.* **2002**, *287*, 337–353. [CrossRef]
7. Bennett, C.H.; Brassard, G.; Crépeau, C.; Jozsa, R.; Peres, A.; Wootters, W.K. Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Phys. Rev. Lett.* **1993**, *70*, 1895–1899. [CrossRef] [PubMed]
8. Buhrman, H.; Cleve, R.; Wigderson, A. Quantum vs. Classical Communication and Computation. In Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, STOC '98, Dallas, TX, USA, 24–26 May 1998; ACM: New York, NY, USA, 1998; pp. 63–68. [CrossRef]
9. Cleve, R.; van Dam, W.; Nielsen, M.; Tapp, A. Quantum entanglement and the communication complexity of the inner product function. *Theor. Comput. Sci.* **2013**, *486*, 11–19. [CrossRef]
10. Buhrman, H.; de Wolf, R. Communication Complexity Lower Bounds by Polynomials. In Proceedings of the Computational Complexity. Sixteenth Annual IEEE Conference, Los Alamitos, CA, USA, 18–21 June 2001; p. 0120. [CrossRef]
11. Razborov, A.A. Quantum communication complexity of symmetric predicates. *Izv. Math.* **2003**, *67*, 145. [CrossRef]
12. Klauck, H. Lower Bounds for Quantum Communication Complexity. *SIAM J. Comput.* **2007**, *37*, 20–46. [CrossRef]
13. van Dam, W.; Hayden, P. Renyi-entropic bounds on quantum communication. *arXiv* **2002**, arXiv:quant-ph/0204093. Available online: http://arxiv.org/abs/quant-ph/0204093 (accessed on 18 October 2024).
14. Marwah, A.; Touchette, D. Optical quantum communication complexity in the simultaneous-message-passing model. *Phys. Rev. A* **2020**, *102*, 062608. [CrossRef]
15. Le Gall, F.; Suruga, D. Bounds on Oblivious Multiparty Quantum Communication Complexity. In Proceedings of the LATIN 2022: Theoretical Informatics, Guanajuato, Mexico, 7–11 November 2022; Casta neda, A., Rodríguez-Henríquez, F., Eds.; Springer: Cham, Switzerland, 2022; pp. 641–657.
16. Gall, F.L.; Nakajima, S. Multiparty Quantum Communication Complexity of Triangle Finding. In Proceedings of the 12th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2017), Paris, France, 14–16 June 2018; Wilde, M.M., Ed.; Leibniz International Proceedings in Informatics (LIPIcs); Schloss-Dagstuhl-Leibniz Zentrum für Informatik: Dagstuhl, Germany, 2018; Volume 73, pp. 6:1–6:11. [CrossRef]
17. Lee, T.; Schechtman, G.; Shraibman, A. Lower Bounds on Quantum Multiparty Communication Complexity. In Proceedings of the 2009 24th Annual IEEE Conference on Computational Complexity, Paris, France, 15–18 July 2009; pp. 254–262. [CrossRef]
18. Briet, J.; Buhrman, H.; Lee, T.; Vidick, T. Multiplayer XOR games and quantum communication complexity with clique-wise entanglement. *arXiv* **2009**, arXiv:0911.4007. [CrossRef]

19. Buhrman, H.; van Dam, W.; Høyer, P.; Tapp, A. Multiparty quantum communication complexity. *Phys. Rev. A* **1999**, *60*, 2737–2741. [CrossRef]
20. Xue, P.; Li, C.F.; Zhang, Y.S.; Guo, G.C. Three-party quantum communication complexity via entangled tripartite pure states. *J. Opt. B Quantum Semiclass. Opt.* **2001**, *3*, 219. [CrossRef]
21. Galv ao, E.F. Feasible quantum communication complexity protocol. *Phys. Rev. A* **2001**, *65*, 012318. [CrossRef]
22. Shannon, C. The zero error capacity of a noisy channel. *IRE Trans. Inf. Theory* **1956**, *2*, 8–19. [CrossRef]
23. Python Code for ILP in Sec. V. Available online: https://github.com/mengruoyu/ILP (accessed on 18 October 2024).