Secure multiparty computation with a dishonest majority via quantum means

Klearchos Loukopoulos[1,*] and Daniel E. Browne[2]

[1]*Department of Materials, University of Oxford, Parks Road, Oxford OX1 4PH, United Kingdom*

[2]*Department of Physics and Astronomy, University College London, Gower Street, London WC1E 6BT, United Kingdom*

(Received 12 June 2009; revised manuscript received 31 December 2009; published 25 June 2010)

We introduce a scheme for secure multiparty computation utilizing the quantum correlations of entangled states. First we present a scheme for two-party computation, exploiting the correlations of a Greenberger-Horne-Zeilinger state to provide, with the help of a third party, a near-private computation scheme. We then present a variation of this scheme which is passively secure with threshold $t = 2$, in other words, remaining secure when pairs of players conspire together provided they faithfully follow the protocol. Furthermore, we show that the passively secure variant can be modified to be secure when cheating parties are allowed to deviate from the protocol. We show that this can be generalized to computations of n-party polynomials of degree 2 with a threshold of $n - 1$. The threshold achieved is significantly higher than the best known classical threshold, which satisfies the bound $t < n/2$. Our schemes, each complying with a different definition of security, shed light on which physical assumptions are necessary in order to achieve quantum secure multiparty computation.

DOI: 10.1103/PhysRevA.81.062336 PACS number(s): 03.67.Dd, 03.65.Ud

I. INTRODUCTION

Secure multiparty computation (SMPC) is an important and well-studied cryptographic protocol. It was originally introduced by Yao [1] in the form of the "millionaire problem," in which two millionaires wish to discover which of them is the richest without revealing the size of their personal fortunes. In its general form, SMPC refers to the case where n parties, each holding a set of private variables, want to compute a publicly available function f without revealing any information about their variables to other parties, beyond that revealed by the output of function itself. Real-world applications of SMPC include market clearing-price scenarios, secure voting, and on-line bidding [2].

It would be simple to achieve SMPC if a trusted third party were available. In this "ideal scenario," each party would securely send their private data to this trusted third party, who would perform the computation privately and then announce the result. Studies of SMPC are therefore concerned with scenarios where no party can be trusted. Such schemes are deemed to be secure (under specified constraints on the computational power and activities of the parties) when the information learned by each party about their neighbors' inputs matches the ideal case.

SMPC has been studied under a variety of assumptions. These may be limitations on the computational power available to adversaries or restrictions on the level to which parties are allowed to deviate from an agreed protocol. A protocol is called "computationally secure" when its security assumes that an adversary lacks the computational power needed to perform a certain computation, which is believed to be untractable. The ability to compute this function would allow him to break the security of the protocol. An example of this kind of security is the well-known RSA public key system, which bases its security on the hardness of factoring large numbers. The downside to computational security is that the schemes may be vulnerable to new algorithms and new technologies, such as quantum computation, where an efficient factoring algorithm [3] is known. A protocol is called "information-theoretically" secure when its security properties hold independent of the computational power of the adversary. An example of an information-theoretically secure cryptographic system is the well-known Vernam cipher [4].

A further ingredient in security assumptions of an SMPC scheme is the allowed behavior of the participants in the protocol. This has been an important component in classical studies of the SMPC computation, but has so far not been focused on in quantum approaches. The most important behavior models occurring in the classical literature are the "passive" security model and the "active" model.

A protocol has "passive security" with "threshold" t, if it remains secure provided all parties follow the protocol exactly, but t or fewer parties are "corrupted." The corrupted parties may form a coalition, sharing data during the execution of the protocol. A protocol is "actively secure" with threshold t if security is retained when parties are allowed to deviate arbitrarily from the protocol and t or fewer parties are corrupted. In particular, parties in an actively secure model may send incorrect data during the protocol in an attempt to trick other parties into revealing extra information about their inputs. In addition to these standard definitions, we call a protocol "private" if security is achieved when all parties follow the protocol and do not share data (equivalent to the $t = 1$ passive case). We call a protocol "nearly private" when less information is revealed about the parties inputs than public computation but the ideal scenario is not quite attained.

The first solutions to the SMPC problem were based on computational security assumptions. These include Yao's solution to the millionaire's problem [1] and more general treatments [5,6]. Later, information theoretic solutions were shown in [7–11]. A summary of the assumptions, thresholds, and efficiency of these protocols can be found in [11]. The best thresholds for these schemes are upper bounded by $n/2$—in other words, an honest majority is required.

[*] klearchos.loukopoulos@seh.ox.ac.uk

After the success of quantum key distribution [12–14], there was a concerted attempt to construct protocols with quantum enhancement for a number of key cryptographic primitives, such as bit commitment [15,16] and SMPC. Bit commitment was shown to be impossible [15,16]; however, quantum protocols were successfully found for quantum secret sharing of classical information [17–20] and quantum information [21,22].

It was thus natural to consider whether quantum advantages may assist in the problem of SMPC. While computational security is possible in the quantum case, for example, by combining trapdoor functions together with the detectable Byzantine agreement protocol proposed by Fitzi *et al.* [23], surprisingly, this advantage was found to be limited for the case of unconditional security. In fact, it was shown by Lo [24] (see [25,26] for recent generalizations) that deterministic two-party-setting computation was impossible, even with quantum means (see [25,26] for recent generalizations of this result). [27]

Here we uncover key assumptions in Lo's seminal theorem through the use of several security models and show which parts of the theorem correspond to different security assumptions. We identify the most general security model which provides unconditional security while remaining compliant with Lo's theorem and construct a protocol which satisfies this model.

We introduce a "quantum passive" security model, a variant of the passive security model well-studied in the classical case. A key assumption in Lo's and Colbeck's no-go theorems is that the entire protocol may be modeled by a "unitary black box." In the quantum passive model, this assumption does not hold. Under this model, we offer a quantum solution to the SMPC problem by presenting a protocol which is secure against external eavesdroppers and coalitions between party members within the quantum passive secure model which we introduce. Furthermore, we introduce a no quantum cheating channel (NQCC) model, which allows corrupted parties to lie or deviate from the protocol and prove that our protocol offers a solution to the SMPC problem compliant with NQCC security.

Our schemes exploit the nonclassical correlations of Greenberger-Horne-Zeilinger (GHZ) states [28–32], recently shown [33] to be resources for classical computation similar to the way cluster states are a resource for universal quantum computation [34,35]. The advantage provided by our quantum schemes is that, for certain functions, it provides a higher security threshold than all current classical schemes, remaining secure even when all but one party are dishonest.

The structure of this article is as follows: In Sec. II we briefly review Anders and Browne's reinterpretation of the well-known GHZ quantum correlation in terms of the computation of the Boolean AND function. In Sec. III we show how this may be developed into a nearly private multiparty computation scheme. In Sec. IV, we perform a privacy analysis on the protocol of Sec. III and discuss its weaknesses.

Section V contains our definition for our quantum security model, and in Sec. VI we then present a variation of the same protocol which is passively secure with threshold $t = 2$. Section VII contains a passive security analysis of our protocol and a proof that it is passively secure, which includes a discussion on how attacks similar to those in [24,25] relate to our scheme and the quantum security model which we introduce. Then, in Sec. VIII we define the NQCC, which is a variation of the passive model where corrupted parties can lie. Section IX offers an upgrade to the passively secure protocol so that it is NQCC secure, and after this, in Sec. X, we present its security analysis.

In Sec. XI we demonstrate that quantum mechanics has the potential to offer a better corruption threshold than classical protocols, indeed for certain function classes, it can become maximal; that is, $t = n - 1$ for an n-member party.

II. GREENBERGER-HORNE-ZEILINGER CORRELATIONS FOR DISTRIBUTED COMPUTATION AND SECRET SHARING

In this section we briefly review the recent work [33] in which the correlations present in measurements upon the GHZ state are interpreted as a distributed computation of the Boolean AND function. This can be seen most clearly by considering the stabilizer equations for the GHZ state, first presented by Mermin [31],

$$\begin{aligned}\sigma_z \otimes \sigma_z \otimes \sigma_z \,|\psi\rangle &= |\psi\rangle,\\ \sigma_z \otimes \sigma_x \otimes \sigma_x \,|\psi\rangle &= |\psi\rangle,\\ \sigma_x \otimes \sigma_z \otimes \sigma_x \,|\psi\rangle &= |\psi\rangle,\\ \sigma_x \otimes \sigma_x \otimes \sigma_z \,|\psi\rangle &= -\,|\psi\rangle,\end{aligned} \tag{1}$$

where for notational convenience later in this article we have chosen the locally equivalent GHZ state $|\psi\rangle = (1/\sqrt{2})(|y_- y_- y_+\rangle + |y_+ y_+ y_-\rangle)$, with $|y_+\rangle = (1/\sqrt{2})(|0\rangle + i\,|1\rangle)$ and $|y_-\rangle = (1/\sqrt{2})(|0\rangle - i\,|1\rangle)$. This is locally equivalent to the more well-known GHZ state $(1/\sqrt{2})(|000\rangle + |111\rangle)$ and inability for these equations to be simultaneously satisfied by c-number scalar values, representing the measured value in a hidden variable theory, is sometimes known as the GHZ paradox.

We imagine that the three qubits are divided among three parties, each of which will measure in either the σ_z or the σ_x basis and label these measurement operators $O_0 = \sigma_z$ and $O_1 = \sigma_x$. We can then rewrite the four preceding equations in compact form,

$$O_a \otimes O_b \otimes O_{a\oplus b}\,|\psi\rangle = (-1)^{\mathrm{AND}(a,b)}\,|\psi\rangle, \tag{2}$$

where a and b are bit values and $\oplus$ denotes addition modulo 2.

Note that the value of the Boolean AND of bits a and b is encoded in the eigenvalues of these equations. Representing the measured eigenvalues $+$ and $-$ with the bit values $M_i \in \{0,1\}$, we see

$$M_1 \oplus M_2 \oplus M_3 = \mathrm{AND}(a,b). \tag{3}$$

We therefore see that if three parties sharing $|\psi\rangle$ make measurements determined by bit values a, b and $a \oplus b$, the parity of their output bits is equal to AND(a,b). An interesting aspect of this is that the computation can be done in a distributed manner. The qubits which form the GHZ state do not have to be in the same spacial point; they can be distributed between spatially separated parties. The outcome of

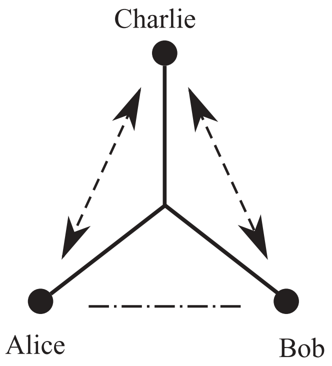

FIG. 1. In this figure we present graphically our scheme. The three party members, Alice, Bob, and Charlie, are connected with straight lines which represent the GHZ state. The dot-dashed line corresponds to the shared randomness resource (e.g., a Bell state) which is shared between Alice and Bob and the dashed lines, whose ends are arrows, that exist between Charlie and Alice and between Charlie and Bob represent a secure classical channel.

the computation is naturally encoded in the parity of bits held by the three parties. This is a simple form of secret sharing [36] because the value is only revealed if the three parties share their data. In the next sections, we use this property as the basis of protocol for SMPC.

III. SCHEME A: A NEARLY PRIVATE MULTIPARTY COMPUTATION PROTOCOL

In this section, we introduce a scheme for multiparty computation between two parties, Alice and Bob. To circumvent Lo's [24] no-go theorem, we add a third party, Charlie. Adding a third party allows a measurement-based scheme to be employed and this introduces an irreducibly classical part, the measurement outcomes, into the protocol. It is this classical part which makes the computation model differ from the "unitary black box" model used in no-go theorems [24,25]. The scheme has enhanced privacy compared to public computation, but Charlie learns more information about Alice and Bob's input than in the ideal scenario. We therefore call this scheme "nearly private." To implement the scheme, Alice, Bob, and Charlie must share a GHZ state $|\psi\rangle$ and additionally, each pair must share secret correlated random bits. In addition, Charlie is able to send data on a secure classical channel to Alice and also to Bob, as shown in Fig. 1. The correlated private bits and the secure channel can both be achieved in information-theoretically secure manner by standard quantum key distribution protocols [12–14].

Let $f(x_1,\dots,x_n,y_1,\dots,y_n)$ be the function to be calculated and let $\vec{x}=(x_1,\dots,x_n)$ and $\vec{y}=(y_1,\dots y_n)$ represent Alice and Bob's input data. In order to simplify our protocol, we make use of the fact that any Boolean function $f(\vec{x},\vec{y}):\{0,1\}^n\times\{0,1\}^n\rightarrow\{0,1\}$, $\vec{x},\vec{y}\in\{0,1\}^n$ can be calculated as the inner product of two vectors of polynomials, $P_i(\vec{x})$ and $Q_i(\vec{y})$ [37],

$$f(x_1,..,x_n,y_1,..,y_n)=\bigoplus_{i=1}^{m}P_i(\vec{x})Q_i(\vec{y}), \tag{4}$$

where the sum operator corresponds to addition modulo 2. The number of terms that will be needed, m, is, at worst, bounded by 2^n, where n is the length of the input vectors; therefore, this decomposition can only be practically employed when m scales polynomially with n, $m\sim\text{poly}(n)$. The function can be evaluated by the calculation of each product P_iQ_i in turn. The polynomials P_i and Q_i can be calculated locally, and hence privately by Alice and Bob respectively. All that is required in addition is the ability to compute $P_iQ_i=\text{AND}(P_i,Q_i)$ for each value of i. Our first protocol exploits the correlations of a GHZ state to achieve this.

The protocol proceeds in the following steps, all addition is performed modulo 2:

1. Repeat steps 2–7 for all the terms present in Eq. (4), starting with $i=1$.
2. Alice and Bob calculate P_i and Q_i locally.
3. Alice and Bob generate a private shared random bit r_i, for example, by suitable measurements on a Bell state, or via a secure communications channel.
4. Alice transmits to Charlie the bit value $P_i\oplus r_i$.
5. Bob transmits to Charlie the bit value $Q_i\oplus r_i$.
6. Charlie adds together these bits to reconstruct $P_i\oplus Q_i=(P_i\oplus r_i)\oplus(Q_i\oplus r_i)$.
7. Alice, Bob, and Charlie measure their qubits of the GHZ state $|\psi\rangle$ as determined by bit values P_i, Q_i, and $P_i\oplus Q_i$; for bit value 0, they measure σ_z; and for bit value 1, they measure σ_x.
8. Once this has been completed for all i terms, Alice and Bob sum their local measurement outcomes. They send their summation bits to Charlie, who sums them with his own measured outcomes.
9. Charlie reveals the value of $f(\vec{x},\vec{y})$.

The correctness of the protocol follows simply from the analysis in the previous section. Due to the correlations of the GHZ state, the value of P_iQ_i is encoded in shared secret form across the three parties. We analyze the privacy in the following section.

IV. PRIVACY ANALYSIS OF SCHEME A

In this section we examine scheme A, step by step, and identify how much each party learns about the inputs of Alice and Bob at each stage.

In steps 1–3 all operations are local; therefore, no information about the private bits can be obtained. After steps 4–6, Charlie receives the parity of Alice and Bob's private bits P_i and Q_i. By use of private channels, no third party could learn these bit values. By use of random bit r, Charlie does not learn anything about the individual values P_i and Q_i other than their parity. In steps 7–9 the three parties exchange bits which, individually, carry no information about either the input or the outputs of the function. After the value of the function is announced, Alice has learned nothing about Bob's inputs more than she would in the ideal scenario. Bob has learned a similar amount about Alice's inputs.

The only deviation from the ideal scenario is the parity information for each term learned by Charlie. This information could, under certain circumstances, be used by Charlie to reconstruct some of Alice and Bob's input data. As an extreme example of this, consider the two-input function

$f(x_1,y_1) = x_1y_1$. If the function outcome is 0 and the parity of the bits is even, Charlie knows with certainty that both Alice and Bob's inputs were 0. We therefore say this protocol is nearly but not completely private.

Since Charlie knows the parity of Alice and Bob's P_i and Q_i bits at every stage, the protocol is intrinsically insecure against any coalition. If Charlie and Bob collaborate, for example, they can learn all of Alice's P_i bits. The protocol, and indeed any protocol where Charlie learns similar parity information, is therefore not passively secure above a trivial threshold $t = 1$. In Sec. VI, we modify the preceding scheme such that Charlie never learns such parity information, and in doing so, introduce a scheme which is passively secure.

V. MODEL FOR PASSIVE SECURITY

Passive security is an important security model in classical SMPC. In this section we introduce a variant "quantum passive security," a generalization of the classical model, where the participants' behavior with respect to quantum resources is specified.

In classical passive security, which we again summarize in what follows, corrupted parties can exploit the information they gain throughout the execution of an SMPC protocol, even collaboratively by forming coalitions, but do not deviate from the protocol. In classical SMPC this means that corrupted parties can exchange classical information and use the total information they infer to learn the private data of honest parties; this approach is also known as "honest but curious."

In our quantum passive security model, similarly to the classical models, corrupted adversaries are allowed to exchange classical information; however, they are not permitted to exchange quantum data and do not possess any shared quantum resources additional to the GHZ states provided through the protocol. In general, we consider SMPC protocols that are n-sided; that is, all parties learn the computation outcome. However, we discuss a one-sided variation of our protocol in Sec. VII to show that it parries so-called EPR attacks.

The preceding are summarized in the following table:

Property	Classical passive security	Quantum passive security
Private data are not inferred by corrupted parties	✓	✓
Corrupted parties do not lie—their outputs are true the protocol	✓	✓
Corrupted parties exchange classical information	✓	✓
Corrupted parties exchange quantum information	N/A	×

The restriction on the exchange of quantum information is important. It is this assumption which means that Lo and Colbeck's unitary black-box models (and hence their no-go theorems) do not apply. In what follows, we explain how allowing such communication does indeed break the presented protocol.

VI. SCHEME B: PASSIVELY SECURE MULTIPARTY COMPUTATION PROTOCOL

In the previous section, we saw that scheme A is prevented from fulfilling requirements for passive security by Charlie's knowledge of the parity of Alice and Bob's input bits. Here, we extend the protocol described in Sec. III and enhance it in order to make it passively secure, that is, secure in the case where party members create coalitions and are allowed to share their data but still do not deviate from the protocol. It is clear that we must modify the protocol such that Charlie never learns the value of the parity of P_i and Q_i bits. Initially, this seems problematic, since Charlie needs this parity information to perform the needed measurement, as described in Sec. II. We can avoid this if we allow Alice and Bob to prepare the entangled states used in a special way, "padding" them with additional random Hadamard tranformations $H = \frac{|0\rangle+|1\rangle}{\sqrt{2}}\langle 0| + \frac{|0\rangle-|1\rangle}{\sqrt{2}}\langle 1|$ known only to themselves. If the parity bit received by Charlie is similarly padded, he can perform the required measurement on this state without ever learning the parity value.

The protocol thus proceeds as follows:

1. Repeat steps 2–11 for all terms present in Eq. (4), starting with $i = 1$.

2. Alice and Bob will calculate P_i and Q_i locally.

3. Alice and Bob each generate a local random "preparation pad" bit, p_a and p_b.

4. Acting simultaneously, the three parties cyclically permute their qubits. Charlie gives his qubit to Alice, Alice gives hers to Bob, and Bob gives his to Charlie.

5. If $p_a = 1$, Alice applies a Hadamard to Charlie's original qubit.

6. They cyclically permute the qubits again, Alice to Bob to Charlie to Alice.

7. Now Bob possesses Charlie's original qubit. If $p_b = 1$, Bob applies a Hadamard to this qubit.

8. They cyclically permute the qubits again, and each regains their initial qubit. (Note that the scheme can be amended such that no quantum communication is needed during the protocol—see later in this article.)

9. The state held by the parties is now $\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b} |\psi\rangle$.

10. Alice and Bob privately give Charlie the bit values $P_i \oplus p_a$ and $Q_i \oplus p_b$, respectively.

11. Alice, Bob, and Charlie measure their qubits in bases according to bit values, P_i, Q_i, and $P_i \oplus Q_i \oplus p_a \oplus p_b$.

12. When all i terms are completed, Alice and Bob sum their local measurement outcomes. They send their summation bits to Charlie, who sums them with his own measured outcomes.

13. Charlie reveals the value of $f(\vec{x},\vec{y})$.

This scheme can be shown to be passively secure with a threshold $t = 2$.

To see that the scheme produces the correct output, note that Charlie's measurement is in the basis $H^{P_i \oplus Q_i \oplus p_a \oplus p_b} Z H^{P_i \oplus Q_i \oplus p_a \oplus p_b}$. This measurement is equivalent to first applying the operator $H^{p_a \oplus p_b}$, and thus effectively undoing the extra Hadamards applied by Alice and Bob, and then performing a measurement determined via bit value $P_i \oplus Q_i$. Thus, the output of the protocol is equivalent to scheme A.

We show in the next section that the scheme is passively secure with threshold $t = 2$. The scheme, as described previously, has the undesirable feature that Charlie's qubit needs to be transmitted coherently between Charlie, Alice, and Bob. However, this is not necessary. Instead, the parties could prepare an ensemble of five-qubit states:

$$\rho = \sum_{p_a,p_b} |p_a\rangle\langle p_a| \otimes |p_b\rangle\langle p_b| \otimes (\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b})\, |\psi\rangle\langle\psi|\, (\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b}). \quad (5)$$

The first qubit is held by Alice, the second by Bob, the latter three by Alice, Bob, and Charlie, respectively. Instead of generating preparation pads, Alice and Bob simply measure their first qubits in the computational basis and use the outcomes as their pad bits. The state of the remaining qubits is then already $\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b} |\psi\rangle$. This replaces steps 3–8 of the protocol and Alice, Bob, and Charlie can continue the protocol from step 9.

VII. PASSIVE SECURITY ANALYSIS OF SCHEME B

In this section, we analyze scheme B step by step, considering for each step, the information which each party learns about the others' private data and the information which they will gain if they form a coalition. We assume at each point that the parties follow the protocol precisely; thus, any cheating is restricted to additional (classical) communication between corrupted parties. This is the standard setting for passive security in the SMPC literature.

In steps 1–3 no data is shared, thus no information can be learned by the parties in any case. In steps 3–9 the three parties cyclically permute their qubits, and Alice and Bob apply local transformations dependent on their private data. Since we are assuming that the protocol is followed by all parties, they may not measure their qubits. Even if they did make measurements, the local state of each qubit is maximally mixed and no information can be gained from the measurement. It is important that at no time does any party possess two qubits, since, after Alice or Bob have applied their pad-Hadamard, a joint measurement of the padded qubit together with one other qubit can reveal the pad bit. The reason for this is that the full stabilizer set of the GHZ state $|\psi\rangle$ contains bipartite operators such as $-\mathbb{1} \otimes Y \otimes Y$, which transforms to $+\mathbb{1} \otimes Y \otimes Y$ when a Hadamard is applied the second or third qubit. Any applied Hadamard would thus be detectable via a measurement of $\mathbb{1} \otimes Y \otimes Y$. Furthermore, this attack would be hidden to the other party since it does not change the state. In the passive model, we assume that coalitions do not have the power to perform joint measurements, since that would require quantum communication between parties, which is considered an active deviation from the protocol. Thus, in this model, such an attack is disallowed.

In step 10, Charlie receives the bit values $P_i \oplus p_a$ and $Q_i \oplus p_b$. From this information he obtains neither the values of P_i or Q_i nor their parity. In order to obtain this, he would need to obtain the values of p_a and/or p_b. He cannot obtain these values in the previous round, and his sole qubit at this stage is, from his perspective, maximally mixed. Also, he cannot obtain these values by forming a coalition, which would, at best, provide him the private data of the coalition partner. In step 11, Alice, Bob, and Charlie measure their qubits, and learn bit values whose parity encodes the product $P_i Q_i$. This is an example of a "shared secret." All three parties must come together to learn the value of $P_i Q_i$ so this step is again secure, even under coalitions of two parties. In steps 12 and 13, Alice and Bob sum their measured bits and send them to Charlie, who then announces the value of $f(\vec{x},\vec{y})$. This is secure, even under passive coalitions, for the same reason as step 11.

A. EPR-type attacks

EPR-type attacks (named for the seminal Einstein-Podolsky-Rosen article) [24] have a special significance for passively secure quantum SMPC protocols because if a corrupted party can infer information about the honest parties' data without being caught, by using a quantum computer and delaying measurements, the role of passive security would be of reduced value in the context of quantum systems. This is because, essentially, the private data of honest parties can leak even when the corrupted party is just performing local operations.

EPR attacks have been demonstrated in the case of one-sided protocols and while our protocol as presented in Sec. VI is n-sided, a one-sided variant can be easily created if steps 11 and 12 are modified as follows:

1. When all i terms are completed, Alice and Bob sum their local measurement outcomes. Alice sends her summation bit to Charlie, who sums it with his own measured outcomes.
2. Charlie sends the parity of his summed bits to Bob, who calculates the value of $f(\vec{x},\vec{y})$.

The question in such attacks is can Bob learn the function outcomes for many values of his vector $\vec{y}$ without someone noticing? Bob is allowed to perform any quantum operation on his qubit(s) while he is attempting to infer the value of f for Alice's given $\vec{x}$ and many possible $\vec{y}$'s. As proven by Lo in [24], if the entire protocol can be modeled as a unitary black box, then this attack successfully allows Bob to break the protocol, by applying unitary transformations to his part of the Hilbert space, which allow him to "poll" the black box for the output of the function for many input vectors and hence learn information about Alice's input.

The reason why this attack fails in our protocol is that the repeated polling by any party is impossible; parties commit to an input value in two ways: first in the classical bit sent to Charlie in step 10, and second in the unbiased nature of the measurements corresponding to different input values. This means that consecutive polling by any corrupted party is impossible.

If, on the other hand, corrupted parties were allowed to communicate quantumly, Bob could, for example, send his qubit to Charlie. Now Charlie could poll both possible input values. In possessing both qubits, he could make a joint measurement, and the relevant joint measurement pairs ($X \otimes X, Z \otimes Z$ or $X \otimes Z, Z \otimes X$) commute. Thus, if quantum communication were allowed between corrupted parties, the EPR attack would succeed.

This feature is related to the property that GHZ-type paradoxes occur in tripartite but not in bipartite systems and illustrates that it is the inability to model the quantum passive secure model via a unitary black box, which is the key to avoiding the no-go theorem.

VIII. NO QUANTUM CHEATING CHANNEL SECURITY MODEL

We have now shown scheme B to be passively secure. However, there is a problem with the notion of pure passive security in the quantum case. Assuming all parties are perfectly honest provides bit commitment for free, and in combination with the results by Yao [38], where it is proven that quantum bit commitment provides oblivious transfer, and Kilian [39], where is it proven that classical oblivious transfer provides classical SMPC, the definition of passive security itself would imply SMPC. Therefore, we expand our notion of security to include the case where corrupted parties are allowed to lie.

In this section we introduce the notion of No Quantum Cheating Channel (NQCC) security model, where no restriction is imposed on the corrupted parties but the use of a quantum channel. We consider a protocol to be NQCC compliant if, on top of safeguarding the honest party members' data, it can detect attempts to corrupt the procedure, therefore allowing the execution of the protocol to be terminated.

The characteristics of NQCC security are summarized in the following table:

Property	NQCC security
Private data are not inferred by corrupted parties	✓
Corrupted parties are *allowed to lie*	✓
Lying is detected	✓
Corrupted parties exchange classical information	✓
Corrupted parties exchange quantum information	×

The NQCC model is the most general security model, in a measurement-based scheme, which remains compliant with Lo's no-go theorem. Removing the only restriction imposed by this model, that is, allowing the use of quantum channels, would make the system equivalent to a unitary black box and then the attack invented by Lo compromises security. The value of NQCC security is that it sheds light on which parts of Lo's theorem are the most crucial for measurement-based SMPC. By highlighting these parts of the theorem, it could be possible that physical systems can be devised that are compliant with NQCC restrictions. These physical systems would then consist of candidates for implementing SMPC at the quantum level.

IX. SCHEME C: NQCC SECURE PROTOCOL

In this section, we extend scheme B so that it becomes compliant with the NQCC security model. Since in the quantum case the definition of passive security automatically implies SMPC, this extension is essential for the usefulness of our protocol. Furthermore, since NQCC is the most general form of security consistent with Lo's theorem, achieving this security level makes our protocol maximally secure under the restrictions imposed by quantum mechanics.

Since Lo/EPR type attacks are not possible, due to restrictions imposed by the model, corrupted parties cannot learn the honest member's data but they can try to misinform him about the output by providing $1 \oplus L$, where L is the sum of their local measurements, during step 12 of scheme B. If they can do so successfully, then the honest parties learn a false value of the outcome but the dishonest party member will learn the correct outcome.

In scheme C this will be detected as follows: If one party member, for example, Alice, artificially sets all her P_i's equal to zero, $P_i = 0$, then the outcome of the function has to be zero. If Bob or Charlie are bit flipping the sum of their local measurements, this would be revealed, as the function outcome would be nonzero.

Therefore, parties repeat scheme B many times, and in each execution of the protocol Alice and Bob would have a probability according to which they set all their P_i's and Q_i's, respectively, equal to zero. Instead of privately giving to Charlie the sum of their measurements, they announce it, and along with that they announce if they were measuring as security testers. Since many repetitions of a scheme B are required, in order to detect cheating, we introduce one more index, j, which enumerates repetitions of scheme B.

Scheme C can be summarized in the following steps:

1. Agree on a number of repetitions, N_{rep}, and have Alice and Bob choose their probabilities to act as security testers, $t_a < 0.5$ and $t_b < 0.5$, respectively. Alice and Bob may choose the probabilities $0 < t_a < 1$, $0 < t_b < 1$ during the execution of protocol.
2. For $j = 1$ to $j = N_{\text{rep}}$ repeat the following steps:
3. Repeat steps 4–13 for all terms present in Eq. (4), starting with $i = 1$.
4. Alice and Bob will calculate P_i and Q_i locally. According to the values of t_a and t_b, they may choose to set $P_i = 0$, $Q_i = 0$.
5. Alice and Bob each generate a local random "preparation pad" bit, p_a and p_b.
6. Acting simultaneously, the three parties cyclically permute their qubits. Charlie gives his qubit to Alice, Alice gives hers to Bob, and Bob gives his to Charlie.
7. If $p_a = 1$, Alice applies a Hadamard to Charlie's original qubit.
8. They cyclically permute the qubits again, Alice to Bob to Charlie to Alice.
9. Now Bob possesses Charlie's original qubit. If $p_b = 1$, Bob applies a Hadamard to this qubit.
10. They cyclically permute the qubits again, and each regains their initial qubit. (Note that the scheme can be amended such that no quantum communication is needed during the protocol—see later in this article.)
11. The state held by the parties is now $\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b} |\psi\rangle$.
12. Alice and Bob privately give Charlie the bit values $P_i \oplus p_a$ and $Q_i \oplus p_b$, respectively.
13. Alice, Bob, and Charlie measure their qubits in bases according to bit values, P_i, Q_i, and $P_i \oplus Q_i \oplus p_a \oplus p_b$.
14. When all i terms are completed, Alice, Bob, and Charlie sum their local measurement outcomes. They all concurrently announce their summation bits and also announce if they were acting as security testers for the current j.

15. Everyone calculates the value of $f(\vec{x},\vec{y})$.

16. If either Alice or Bob (or both) announced that they acted as security testers and $f(\vec{x},\vec{y}) \neq 0$, parties halt the protocol. An attempt to compromise it is detected.

17. If neither Alice nor Bob announced that they acted as security testers and the value of $f(\vec{x},\vec{y})$ is inconsistent with values for previous j's when again both were not security testers, the protocol is halted and an attempt to compromise it is detected.

Again, the qubit-swapping can be avoided if instead of a GHZ state the following five-qubit ensemble is shared between the three parties:

$$\rho = \sum_{p_a,p_b} |p_a\rangle\langle p_a| \otimes |p_b\rangle\langle p_b| \otimes (\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b})\,|\psi\rangle\,\langle\psi|\,(\mathbb{1} \otimes \mathbb{1} \otimes H^{p_a \oplus p_b}). \quad (6)$$

X. NQCC SECURITY ANALYSIS OF SCHEME C

Here, we examine scheme C and prove that it is compliant with NQCC security, that is, the private data of honest parties remain uncompromised and cheating attempts are detected. The only assumption we make is that no quantum channel is used.

Since the case of honest but curious corrupted parties was discussed in Sec. VII, here we focus on attacks where corrupted parties deviate from the protocol. Except entering their private data feed to the protocol, parties dynamically interact (provide input) with the protocol, at steps 7 and 9, step 12, and step 14. The attacks which can be generated in these steps are the following:

1. Alice or/and Bob provide to Charlie invalid bit values for $P_i \oplus p_a$ or $Q_i \oplus p_b$.

2. Alice or Bob lie about the sum of their measurement bits.

3. Alice or Bob are dishonest on whether they acted as security testers.

A. Provide to Charlie invalid bit values for $P_i \oplus p_a$ or $Q_i \oplus p_b$

In this attack, Alice and/or Bob lie to Charlie and use fake preparation pads p_a and/or p_b, respectively, which would lead Charlie to use an incorrect measurement axis. In this case, when Charlie is measuring along an incorrect axis, Charlie's measurement outcome would be 1 with 50% probability and 0 with 50% probability. This is something detectable during step 17, as there will be inconsistency in the calculated function values between different protocol runs.

B. Lie about the sum of their measurement bits

Here, one party member is providing a bit-flipped sum of his local measurements. This leads the other party members to learn a wrong (bit-flipped) function value, while the corrupted member would be able to recover the correct value. This compromising strategy has to be followed in every run of the protocol or else, due to step 17, it will be detected. However, if, without loss of generality, Bob is performing this attack, he will be detected when Alice acts as a security tester at a run, during which he does not have the role of a security tester. This detection has a probability of happening on a repetition j of the protocol equal to $t_a(1-t_b)$ and it happens on average after $\frac{1-t_b}{t_a}$ steps.

C. Be dishonest on whether they acted as security testers

Here a party member can either act as a security tester, enforcing a $f=0$ output, without announcing it, or they could announce they acted as security testers without having set their input equal to zero. If the correct function outcome is $f=0$, this attack does not affect by any means the protocol, and if the correct outcome is $f=1$, it will be detected during step 16.

Scheme C is therefore NQCC secure and the security threshold, $t=2$, remains the same since the detection methods work as long as all party members are not corrupted.

XI. ADVANTAGE OVER CLASSICAL SCHEMES

So far, we have presented a scheme for two-party computation which is passively secure with threshold $t=2$. Compared to classical schemes, our scheme has the disadvantage that quantum resources and an extra player are needed. However, by scaling up the scheme to multiparty computation over n parties, a significant advantage of the quantum scheme is revealed.

Known passively secure classical schemes [7,11,40,41] require, in the general case, an honest majority; in other words, their threshold has an upper bound $t < n/2$. By modifying our scheme B, we can construct a scheme which allows secure multipartite computation over a restricted family of n-party functions with a threshold at its maximum value, $t = n-1$. The family of functions we consider are most easily described by considering $f(\vec{x},\vec{y})$ as a polynomial over $\mathbb{Z}_2$ (i.e., where AND represents multiplication and XOR represents addition). They are polynomials of degree 2 which have the following form:

$$f = \bigoplus_{j_1>j_2} \bigoplus_{i} \lambda_{j_1,j_2} P_i^{j_1} P_i^{j_2}, \quad (7)$$

where the j_1, j_2 indices are used to distinguish the n parties and λ_{j_1,j_2} is a bit number which indicates if a pair of parties (j_1, j_2 has a joint computation which is needed in order to evaluate f. As far as we are aware, there is no proof that a classical protocol for secure computation of degree 2 functions requires an honest majority; however, our quantum protocol provides the highest possible corruption threshold. There exist recent examples (e.g., voting) of limited classical SMPC protocols which do not require an honest majority [42]. We hope that our result motivates more work in this area.

Each term in the sum depends on input bits from two parties only, and hence scheme B can be adapted to provide a fully secure computation method. Notably, the threshold for this scheme will remain $n-1$.

The scheme progresses as follows:

1. Repeat steps 2 and 3 for each (j_1,j_2) term in Eq. (7).

2. Parties j_1 and j_2 nominate a third party j_3. These three parties share a GHZ state.

3. The three parties follow scheme B up to step 11. The parties retain their measured bits which are not yet shared.

4. After this has been completed for every participating pair, each party computes the parity of their measured bits and announces the sum.

5. The players compute function f by summing these public bits.

In the n party scheme, the security of the whole computation depends on the security of each of the three party computations performed. Since at each stage prior to the final announcements each term in the sum (7) is encoded in the parity of bits held by parties j_1, j_2, and j_3, each party's input data remain secure, even if all other $n-1$ other parties share data. For this reason, this scheme is passively secure with a threshold of $t=n-1$. Furthermore, if instead of scheme B, the variant discussed in Sec. X is used, then the scheme is NQCC secure, again with $t=n-1$.

XII. DISCUSSION AND CONCLUSIONS

In this article we have introduced a scheme for secure n-party computation. The scheme exploits the intrinsically quantum correlations of the Greenberger-Horne-Zeilinger states to provide security and privacy. By considering a novel security model, inspired by the passive secure settings so important in classical secure computation, we show that unconditionally SMPC may be enhanced by access to a quantum resource. The scheme we present depends upon the natural secret-sharing characteristics of GHZ correlations. Illustrating the potential of such correlations for private computation with a "nearly private scheme," we then upgraded this nearly private scheme to a scheme (scheme B) which is secure under the conditions of quantum passive security defined in Sec. V. Afterward, the protocol was further upgraded to NQCC security, where the parties are allowed to deviate from the protocol, as long as they do not use quantum communication. This was then extended to a scheme for secure n-party computation with a threshold of $n-1$. This n-party scheme is restricted to quadratic functions, but achieves a security threshold higher than any known classical scheme.

The (nonphysical) bipartite object with analogous correlations to the GHZ state [33] is the Popescu-Rohrlich nonlocal box [43]. Thus the nonlocal box (if it existed) would have a further application for secure computation. This observation was made independently very recently [44] and used to calculate better bounds on the number of oblivious transfer calls needed for secure computation of a function of previous estimates [45].

Since the block on quantum communication between cheating players seems the key assumption which allows the security models we discussed to differentiate from the unitary black-box model where the Lo-Colbeck no-go theorems apply, in order to physically realize quantum SMPC, models which fulfill this assumption need to be further researched, for example, noisy quantum storage models, which have recently been shown to have some favorable cryptographic properties [46].

One limitation of the scheme is its restriction to degree 2 polynomials. Nevertheless, even for this restricted class of functions there is no known classical secure scheme which does not require an honest majority. Proving upper bounds on the security of classical schemes for restricted functions would be an interesting research direction, in which we are not aware that any work has been carried out. A recent generalization [47] of [33] to higher-degree functions may provide the means to extend our scheme to higher-degree functions. It is possible that other families of functions with particular symmetries and structure are well suited to this kind of method. A further limitation is the restriction to families of functions, which when written in the form of (7), have a number of terms polynomial in the input size. This appears to be a fundamental limitation of employing the inner-product decomposition [37] since it can be shown that certain functions (e.g., the equality of two bit strings) require exponentially many terms (see [37] for a fuller discussion).

It is natural to ask whether the schemes we have presented can be developed into schemes for secure quantum computation, using cluster states [34] in place of the GHZ states. In fact, it has already been shown that cluster-state-based quantum computation has promising security features, since a secure method of "blind quantum computation" [48] has been developed which utilizes on cluster states measurement-based quantum computation. In light of this, the application of cluster states to secure quantum multiparty computation seems a promising direction.

The development of quantum key distribution has been one of the most successful aspects of quantum information science and is certainly the aspect closest to real-world application. We hope that this work demonstrates that quantum methods can provide advantages in other cryptographic problems and inspires further study in this area.

ACKNOWLEDGMENTS

We thank Daniel Gottesman and Joe Fitzsimons for valuable suggestions. Also, we are thankful to Brendon Lovett, Earl Campbell, and Matthew Hoban for helpful discussions. This work was supported by QIPIRC programme of EPSRC and the National Research Foundation, Singapore and Ministry of Education, Singapore. K. Loukopoulos acknowledges financial support from Materials Department, University of Oxford, and St. Edmund Hall.

[1] A. C. Yao, in *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science* (IEEE Computer Society, Washington, DC, 1982), p. 160.

[2] P. Bogetoft *et al.*, in *Lecture Notes in Computer Science*, edited by R. Dingledine and P. Golle (Springer-Verlag, Berlin/Heidelberg, 2009), Vol. 5628.

[3] P. Shor, SIAM J. Sci. Stat. Comput. **26**, 1484 (1997).

[4] G. S. Vernam, J. IEEE **55**, 109 (1926).

[5] O. Goldreich, S. Micali, and A. Wigderson, in *Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1987), p. 218.

[6] D. Chaum, I. Damgård, and J. van de Graaf, in *Lecture Notes in Computer Science* (Springer-Verlag, London, 1987), Vol. 293, p. 87.

[7] M. Ben-Or, S. Goldwasser, and A. Wigderson, in *Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1988), p. 1.

[8] D. Chaum, C. Crépeau, and I. Damgård, in *Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1988), p. 11.

[9] T. Rabin and M. Ben-Or, in *Annual ACM Symposium on Theory of Computing* (ACM, New York, NY, 1989), p. 73.

[10] D. Beaver, in *Lecture Notes in Computer Science* (Springer-Verlag, London, 1989), Vol. 435, p. 560.

[11] I. Damgård and J. B. Nielsen, in *Lecture Notes in Computer Science* (Springer-Verlag, Berlin/Heidelberg, 2007), Vol. 4622, p. 572.

[12] C. H. Bennett and G. Brassard, in *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing, Bangalore, India* (IEEE, New York, 1984), p. 175.

[13] S. Wiesner, Sigact News **15**, 78 (1983).

[14] A. K. Ekert, Phys. Rev. Lett. **67**, 661 (1991).

[15] D. Mayers, Phys. Rev. Lett. **78**, 3414 (1997).

[16] H. K. Lo and H. F. Chau, Phys. Rev. Lett. **78**, 3410 (1997).

[17] M. Hillery, V. Bužek, and A. Berthiaume, Phys. Rev. A **59**, 1829 (1999).

[18] A. Karlsson, M. Koashi, and N. Imoto, Phys. Rev. A **59**, 162 (1999).

[19] L. X., G. L. Long, F. G. Deng, and J. W. Pan, Phys. Rev. A **69**, 052307 (2004).

[20] K. Chen and H.-K. Lo, Quantum Inf. Comput. **7**, 689 (2007).

[21] D. Markham and B. C. Sanders, Phys. Rev. A **78**, 042309 (2008).

[22] R. Cleve, D. Gottesman, and H.-K. Lo, Phys. Rev. Lett. **83**, 648 (1999).

[23] M. Fitzi *et al.*, in *Annual ACM Symposium on Principles of Distributed Computing* (ACM, New York, NY, 2002), p. 118.

[24] H.-K. Lo, Phys. Rev. A **56**(2), 1154 (1997).

[25] R. Colbeck, Phys. Rev. A **76**, 062308 (2007).

[26] L. Salvail, C. Schaffner, and M. Sotakova, in *Advances in Cryptology—ASIACRYPT 2009*, edited by M. Matsui (Springer, Berlin/Heidelberg, 2009), p. 70.

[27] One way to categorize SMPC protocols is through the number of parties that learn the computation outcome. If one party learns the outcome, then the protocol is called one-sided, if two parties learn the outcome, then the protocol is two-sided, and, more generally, if n parties learn the outcome, then the protocol is n-sided. Our protocol is presented as an n-sided protocol; however, the other variations may be obtained by minor modifications, and we therefore do not emphasize this categorization here.

[28] D. M. Greenberger, M. A. Horne, and A. Zeilinger, in *Bell's Theorem, Quantum Theory, and Conceptions of the Universe*, edited by M. Kafatos (Springer, Berlin, 1989), p. 73.

[29] D. M. Greenberger, M. A. Horne, A. Shimony, and A. Zeilinger, Am. J. Phys. **58**, 1131 (1990).

[30] D. M. Greenberger, M. A. Horne, and A. Zeilinger, Phys. Today **46**(8), 22 (1993).

[31] N. D. Mermin, Am. J. Phys. **58**, 731 (1990).

[32] N. D. Mermin, Phys. Today **43**(6), 9 (1990).

[33] J. Anders and D. E. Browne, Phys. Rev. Lett. **102**, 050502 (2009).

[34] R. Raussendorf and H. J. Briegel, Phys. Rev. Lett. **86**, 5188 (2001).

[35] R. Raussendorf, D. E. Browne, and H. J. Briegel, Phys. Rev. A **68**, 022312 (2003).

[36] A. Shamir, Commun. ACM **22**, 612 (1979).

[37] W. Van Dam, *Implausible Consequences of Superstrong Nonlocality*, e-print arXiv:quant-ph/0501159v1; see also W. van Dam, Ph.D. thesis, University of Oxford, Department of Physics, 2000.

[38] A. C.-C. Yao, in *Proceedings of the 26th Annual ACM Symposium on the Theory of Computing* (ACM, New York, 1995), p. 67.

[39] J. Kilian, in *Proceedings of 1988 ACM Symposium on the Theory of Computing* (ACM, Chicago, 1988), p. 20.

[40] R. Cramer *et al.*, in *Lecture Notes in Computer Science* (Springer, Berlin/Heidelberg, 1999), Vol. 1592, p. 311.

[41] Z. Beerliova-Trubiniova and M. Hirt, in *Lecture Notes in Computer Science*, edited by S. Halevi and T. Rabin (Springer, Berlin/Heidelberg, 2006), Vol. 3876, p. 305.

[42] A. Broadbent and A. Tapp, *Information-Theoretically Secure Voting Without an Honest Majority*, e-print arXiv:0806.1931v1.

[43] A. Popescu and D. Rohrlich, Found. Phys. **24**, 379 (1994).

[44] M. Kaplan *et al.*, in *Leibniz International Proceedings in Informatics* (Schloss Dagstuhl, 2009), Vol. 4, p. 239.

[45] A. Beimel and T. Malkin, in *Lecture Notes in Computer Science*, edited by M. Naor (Springer, Berlin/Heidelberg, 2004), Vol. 2951, p. 238.

[46] S. Wehner, M. Curty, C. Schaffner, and H.-K. Lo, *How to Implement Two-party Protocols in the Noisy-storage Model*, e-print arXiv:0911.2302v1 [quant-ph].

[47] E. Campbell, M. Hoban, K. Loukopoulos, J. Anders, and D. E. Browne (unpublished).

[48] A. Broadbent, J. Fitzsimons, and E. Kashefi, *Foundations of Computer Science* (IEEE Computer Society, Washington, DC), p. 517.